ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

ΣΧΟΛΗ ΕΦΑΡΜΟΣΜΕΝΩΝ ΜΑΘΗΜΑΤΙΚΩΝ ΚΑΙ ΦΥΣΙΚΩΝ ΕΠΙΣΤΗΜΩΝ

Γραμμική Παλινδρόμηση, Μέθοδοι Ridge και Lasso και Δέντρα Παλινδρόμησης.

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Σταμούλη Ιωάννη

Επιβλέπουσα : Καρώνη Χρυσηίς
Καθηγήτρια Ε.Μ.Π.

Χρυσηίς Καρώνη	Βασίλης Παπανικολάου	Καλλιόπη Παυλοπούλου
Καθηγήτρια Ε.Μ.Π.	Καθηγητής Ε.Μ.Π	ΕΔΙΠ Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2022

Ευχαριστίες

Θα ήθελα να ευχαριστήσω την επιβλέπουσα καθηγήτρια μου κ. Χρυσηίς Καρώνη που με καθοδήγησε κατά την εκπόνηση της διπλωματικής μου εργασίας

Περίληψη

Η εργασία αυτή έχει ως στόχο την μελέτη μοντέλων γραμμικής παλινδρόμησης με την μέθοδο των ελαχίστων τετραγώνων, τις μεθόδους ποινών όπως της Παλινδρόμησης Κορυφογραμμής (Ridge Regression) και της Lasso. Επίσης με μεθόδους μηχανικής μάθησης όπως των δέντρων αποφάσεων και πιο συγκεκριμένα των δέντρων παλινδρόμησης και τέλος με την μέθοδο του αλγορίθμου των τυχαίων δασών (Random Forest).
Στο πρώτο κεφάλαιο θα εστιάσουμε σε έννοιες που θα χρειαστούμε για τις μεθόδους που θα μελετήσουμε και στην συνέχεια θα αναλύσουμε εκτενώς την απλή γραμμική παλινδρόμηση και την πολλαπλή γραμμική παλινδρόμηση με την μέθοδο των ελαχίστων τετραγώνων.
Στο δεύτερο κεφάλαιο της εργασίας θα επικεντρωθούμε στην σημασία που έχει το μοντέλο να είναι ακριβές στην πρόβλεψη του και παράλληλα απλό και κατανοητό. Επίσης θα αναφερθούμε σε μεθόδους επιλογής μεταβλητών του μοντέλου και σε κριτήρια αξιολόγησης και επιλογής του μοντέλου.
Στο τρίτο κεφάλαιο, αφού αναφερθούμε στο πρόβλημα που δημιουργεί το φαινόμενο της πολυσυγγραμικότητας στα μοντέλα που κατασκευάζονται με την μέθοδο των ελαχίστων τετραγώνων, θα μελετήσουμε μεθόδους που λειτουργούν αποτελεσματικότερα όταν τα δεδομένα μας παρουσιάζουν αυτό το φαινόμενο. Δύο από αυτές τις μεθόδους είναι η Παλινδρόμηση Κορυφογραμμής (Ridge Regression) και η μέθοδος Lasso οι οποίες ανήκουν στην οικογένεια των μεθόδων με ποινή. Τέλος θα αναλύσουμε την μέθοδο κατασκευής μοντέλου με την χρήση των δέντρων παλινδρόμησης και των μεθόδων Bagging, του αλγορίθμου των τυχαίων δασών (Random Forest) και Boosting.

Στο τέταρτο και τελευταίο κεφάλαιο θα εφαρμοστούν οι μέθοδοι που αναλύσαμε σε ένα παράδειγμα πραγματικής ζωής, το οποίο είναι η μελέτη της συσχέτισης του μισθού των ποδοσφαιριστών του Ιταλικού Πρωταθλήματος και των στατιστικών τους για την αγωνιστική περίοδο 2021-2022, αλλά και η πρόβλεψη του μισθού ενός ποδοσφαιριστή δεδομένου τα στατιστικά που είχε. Για το πρόβλημα αυτό θα χρησιμοποιηθούν δεδομένα που λήφθηκαν από το understat.com για τα στατιστικά των ποδοσφαιριστών και από το capology.com για τον μισθό τους. Το συμπέρασμα στο οποίο καταλήγουμε από την μελέτη του μοντέλου που κατασκευάσαμε είναι ότι παραδοσιακά στατιστικά

όπως οι ασσίστ και οι πάσες κλειδιά δεν συνεισφέρουν στον μισθό ενός ποδοσφαιριστή ενώ τα γκολ συνεισφέρουν ελάχιστα. Αντίθετα, ο μισθός των ποδοσφαιριστών, καθορίζεται κυρίως από προηγμένα στατιστικά όπως τα αναμενόμενα γκολ xG, το xGChain και το xGBuildup τα οποία είναι στατιστικά που θα επεξηγηθούν εκτενώς στην εργασία. Τέλος, το μοντέλο που κατασκευάσαμε μπορεί να προβλέψει με σχετική ακρίβεια τον μισθό ενός παίκτη δεδομένου τα στατιστικά του. Δύναται να χρησιμοποιηθεί για την απόκτηση ενός παίκτη δελεάζοντάς τον με έναν καλύτερο μισθό ή ακόμα και σε περιπτώσεις ανανέωσης συμβολαίου.

Λέξεις-Κλειδιά: Πολλαπλή Γραμμική Παλινδρόμηση, Μέθοδος Ελαχίστων Τετραγώνων, Πολυσυγγραμικότητα, Επιλογή μεταβλητών, Ίχνος Κορυφογραμμής, Lasso, Δέντρα παλινδρόμησης, Αλγόριθμος των τυχαίων δασών, Bagging

ABSTRACT

The objective of this thesis is the study of models of linear regression that are constructed by using Least Square Regression, models that are built by using regularization techniques such as the Ridge and the Lasso and models that are built by using Decision Trees and to be more precise Regression Trees and the Random Forests algorithm.

In the first part of this thesis we will focus on analyzing concepts that we will use later on in the thesis and also we will study models of linear regression and multiple linear regression that are created by using the method of least squares.

In the second part of the thesis we will focus on the importance of the model to be accurate when it comes to predictive power but also simple and easily interpretable. Also we will study methods and criteria that are used for variable selection and ways to select the best model for our dataset.

In the third part we talk about the phenomenon of multicollinearity and how it affects the models that we analyzed in the first chapter. Afterwards we focus on two methods that are used to create models that are more accurate and overall better to use than the least square method when we encounter the problem of multicollinearity, those two methods are the Ridge Regression and the Lasso which are two regularization methods. In the end we will pivot to the method of Decision Trees, more specifically the Regression Trees and the methods of Bagging, Random Forest and Boosting.

In the fourth and final chapter the methods that are studied and explained will be used in a real world experiment that is the prediction of a footballer's salary that plays in the Serie A by using their statistics of the 2021-2022 season. For this problem the data that we used are from understat.com and capology.com, capology.com was used for the salaries of the players. The conclusion we reached by studying this problem is that the salary of the player does not depend on any traditional metrics such as assists and key passes, depends a little on salary and mostly it depends on "advanced statistics" such as xG (Expected Goals), XGChain and xGBuildup that will be further explained in the thesis. Lastly the model that was created is a model with decent predictive accuracy of the salary of the player by taking into account only their statistics on the field and can be used to judge if it

is worth signing a player by luring him with a better salary or for contract extensions.

Keywords: Multiple linear regression, Least Squares method, Multicollinearity, Variable Selection, Ridge Regression, Lasso, Regression Trees, Random Forests, Bagging

Πίνακας Περιεχομένων

1 Εισαγωγή

1.1Απλό Γραμμικό Μοντέλο

Το απλό γραμμικό μοντέλο παλινδρόμησης αποτελείται από δύο είδη μεταβλητών, την ανεξάρτητη ή αλλιώς επεξηγηματική x και την εξαρτημένη ή αλλιώς απόκρισης y. Τα δύο αυτά είδη μεταβλητών συνδέονται μεταξύ τους με μια γραμμική συνάρτηση παλινδρόμησης που έχει ως στόχο την προσαρμογή μιας ευθείας η οποία περιγράφει όσον το δυνατόν καλύτερα την σχέση μεταξύ της επεξηγηματικής και της μεταβλητής απόκρισης (Καρώνη & Οικονόμου, 2017). Η ευθεία είναι της μορφής:

$$E(y|x) = E(y_x) = \beta_0 + \beta_1 x = \mu_x$$

Τα β_0, β_1 ονομάζονται συντελεστές παλινδρόμησης. Στην σχέση αυτή η μεταβλητή x θεωρείται μη στοχαστική ενώ η y θεωρείται ως μια τυχαία μεταβλητή. Προκειμένου να γίνει η καταλληλότερη προσαρμογή της ευθείας λαμβάνουμε υπόψιν τις n ανεξάρτητες παρατηρήσεις (x_i, y_i) για τις οποίες θεωρούμε ότι δεν υπάρχει κάποιο σφάλμα στις μετρήσεις τους. Με αυτόν τον τρόπο παίρνουμε την εκτίμηση των y_i την οποία την συμβολίζουμε ως $\hat{y}_i$. Τα σημεία (x_i, y_i) συνήθως διαφέρουν από τα $(x_i, \hat{y}_i)$ εκτός αν η σχέση των x_i με τα y_i είναι γραμμική. Η σχέση που περιγράφει τα $\hat{y}_i$ σε σχέση με τα x_i είναι η:

$$\hat{y}_i = \widehat{\beta_0} + \widehat{\beta_1} x_i \qquad (1.1)$$

Τα $\widehat{\beta_0}, \widehat{\beta_1}$ είναι οι εκτιμήσεις των συντελεστών β_0 και β_1 αντίστοιχα. Παράλληλα οι παρατηρήσεις y_i δίνονται απο την σχέση

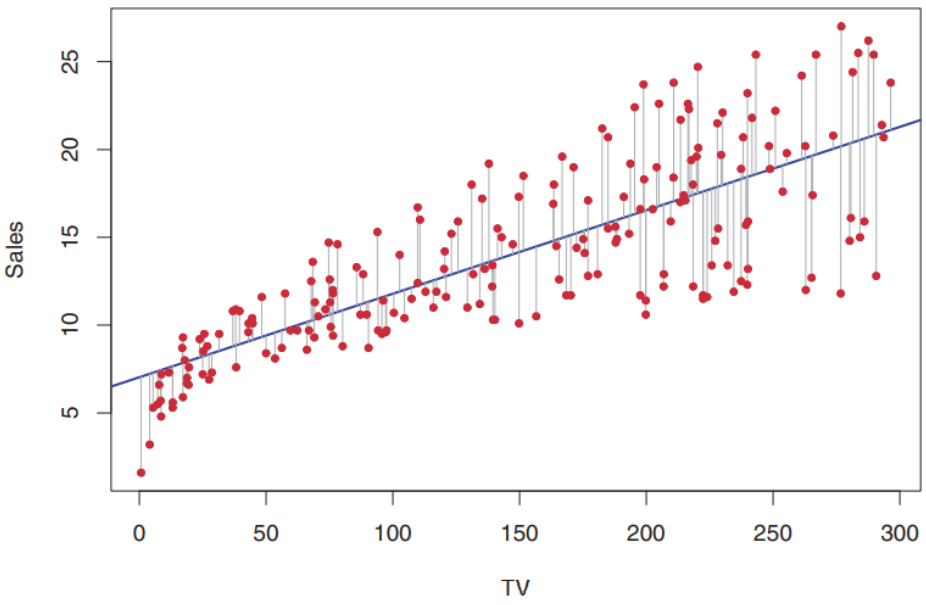
$$y_i = E(y_{x_i}) + \varepsilon_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad (1.2)$$

Το ε_i ονομάζεται τυχαίο σφάλμα και συμβολίζει την κατακόρυφη απόκλιση της τιμής y_i από την εκτίμηση της συνάρτηση παλινδρόμησης.

Επίσης έχουμε τον όρο e_i ο οποίος είναι η κατακόρυφη απόκλιση του y_i από την εκτιμομένη συνάρτηση παλινδρομήσης και δίνεται από τον τύπο

$$e_i = y_i - \hat{y}_i \quad (1.3)$$

Ο όρος αυτός ονομάζεται υπόλοιπο και είναι η εκτίμηση του ε_i. Στο Γράφημα 1 βλέπουμε γραφικά τα e_i που είναι η απόσταση των κόκκινων σημείων απο την εκτιμόμενη συνάρτηση παλινδρόμησης.



Γράφημα 1.1: Γράφημα στο οποίο παρατηρούμε το υπόλοιπο e_i σε ένα απλό μοντέλο που έχει προσαρμοστεί σε ένα γνωστό σύνολο δεδομένων που σχετίζεται με διαφημίσεις. Στο μοντέλο έχει προσαρμοστεί η μεταβλητή Sales (πωλήσεις) στην μεταβλητή TV (τηλεοράσεις). (James et al., 2021, σ.62)

Η μέθοδος με την οποία εκτιμούμε τις ποσότητες $\widehat{\beta_0}$ και $\widehat{\beta_1}$ είναι αυτή των ελαχίστων τετραγώνων. Ο στόχος αυτής της μεθόδου είναι η επιλογή κατάλληλων $\widehat{\beta_0}$ και $\widehat{\beta_1}$ τα οποία ελαχιστοποιούν την ποσότητα RSS (residual sum of squares). Η RSS είναι το άθροισμα των τετραγώνων των αποκλίσεων μεταξύ των y_i *και* $\hat{y}_i$ και δίνεται από την σχέση

$$\text{RSS} = \sum_1^n {e_i}^2 = \sum_{i=1}^n (y_i - \widehat{\beta_0} - \widehat{\beta_1} x_i)^2 \quad (1.4)$$

Προκειμένου να ελαχιστοποιήσουμε την σχέση αυτή θα χρησιμοποιήσουμε την συνάρτηση

$$S(\beta_0, \beta_1) = \sum_{i=1}^n (y_i - \beta_0 - \beta_1 x_i)^2 \quad (1.5)$$

και παίρνουμε τις μερικές παραγώγους του $S(\beta_0, \beta_1)$ ως προς β_0 και β_1 και λαμβάνουμε τις τιμές

$$\widehat{\beta_1} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2} \quad (1.6)$$

$$\widehat{\beta_0} = \bar{y} - \widehat{\beta_1}\ \bar{x} \quad (1.7)$$

Όπου $\bar{x}$, $\bar{y}$ η μέση τιμή των x,y παρατηρήσεων του δείγματος αντίστοιχα.

Τέλος, προκειμένου να αποδείξουμε ότι όντως το σημείο $(\widehat{\beta_0}, \widehat{\beta_1})$ είναι αυτό στο οποίο η συνάρτηση $S(\beta_0, \beta_1)$ λαμβάνει την ελάχιστη τιμή της, θα πρέπει να δείξουμε ότι ο πίνακας των δευτέρων παραγωγών της $S(\beta_0, \beta_1)$ είναι θετικά ορισμένος. Συγκεκριμένα, στην περίπτωσή μας επειδή είναι δύο διαστάσεων η συνάρτηση θα πρέπει από το Κριτήριο δευτέρων παραγώγων για συναρτήσεις δύο μεταβλητών:

- $\frac{\partial S(\widehat{\beta_0}, \widehat{\beta_1})}{\partial \beta_0} > 0$
- det $(S''(\widehat{\beta_0}, \widehat{\beta_1})) > 0$

Έπειτα από πράξεις καταλήγουμε ότι $\frac{\partial^2 S(\widehat{\beta_0},\widehat{\beta_1})}{\partial^2 \beta_0} = 2n > 0$ και det (S¨($\widehat{\beta_0}$, $\widehat{\beta_1}$)) = 4n$\sum_{i=1}^{n}(x_i - \bar{x})^2$ > 0 άρα το ($\widehat{\beta_0}$,$\widehat{\beta_1}$) είναι σημείο τοπικού ελαχίστου και ελαχιστοποιεί την ποσότητα RSS.
Επίσης για να επιβεβαιώσουμε ότι οι συντελεστές $\widehat{\beta_0}$ και $\widehat{\beta_1}$ είναι ακριβείς πραγματοποιούμε έναν στατιστικό έλεγχο t σε ένα 100 (1-α) % διάστημα εμπιστοσύνης με την εξής αρχική υπόθεση για το $\widehat{\beta_0}$ και $\widehat{\beta_1}$ αντίστοιχα

- H_0: $\widehat{\beta_0} = 0$
  H_1: $\widehat{\beta_0} \neq 0$

- H_0: $\widehat{\beta_1} = 0$
  H_1: $\widehat{\beta_1} \neq 0$

και απαιτούμε το p-value του ελέγχου να είναι μικρότερο από α. Αυτό μπορούμε να το κάνουμε καθώς οι συντελεστές ανήκουν στην κανονική κατανομή καθώς και τα τυχαία σφάλματα που δίνονται από την σχέση (1.3) ακολουθούν την κανονική κατανομή κάτι το οποίο είναι μια από τις βασικές αρχές που πρέπει να ισχύουν για να χρησιμοποιήσουμε το απλό γραμμικό μοντέλο, αυτή της κανονικότητας των σφαλμάτων που θα δούμε στην συνέχεια στην Παράγραφο 1.3.

Ακόμα μια σημαντική ποσότητα είναι το Μέσο Άθροισμα Τετραγώνων των υπολοίπων, το οποίο αποτελεί αμερόληπτη εκτιμήτρια της δειγματικής διασποράς σ^2 (Καρώνη & Οικονόμου, 2017) και δίνεται από την σχέση

$$S^2 = \frac{1}{n-2} RSS \qquad (1.8)$$

Το RSS το έχουμε ορίσει στην σχέση (1.4).

1.2 Ακρίβεια του Μοντέλου

Για να εξετάσουμε την ακρίβεια του μοντέλου μας χρησιμοποιούμε δύο ποσότητες, η μια από αυτές είναι το RSE (Residual Standard Error) ή αλλιώς Τυπική Απόκλιση των Υπολοίπων και δίνεται από την σχέση

$$RSE = \sqrt{S^2} = \sqrt{\frac{RSS}{n-2}} \tag{1.9}$$

όπου το RSS και το S^2 είναι τα μεγέθη που έχουν οριστεί στις σχέσεις (1.4) και (1.8) αντίστοιχα. Γενικά όσο μικρότερο είναι το RSE τόσο πιο καλό είναι και το μοντέλο μας γιατί δείχνει ότι η γραμμή παλινδρόμησης είναι προσαρμοσμένη σε ικανοποιητικό επίπεδο με τα δεδομένα μας. Ενώ αν το RSE είναι πολύ μεγάλο σημαίνει ότι η γραμμή παλινδρόμησης έχει μεγάλη απόκλιση από τα δεδομένα μας κατά συνέπεια το μοντέλο μας χρήζει βελτίωσης.

Η άλλη ποσότητα που μας βοηθάει να εξετάσουμε την ακρίβεια του μοντέλου μας είναι ο συντελεστής προσδιορισμού R^2 που δίνεται από την σχέση

$$R^2 = 1 - \frac{RSS}{TSS} \tag{1.10}$$

όπου το RSS έχει οριστεί από την σχέση (1.4) και το TSS (Total Sum of Squares) ή αλλιώς συνολικό άθροισμα τετραγώνων δίνεται από την σχέση

$$\text{TSS} = \sum_{i=1}^{n}(\text{y}_\text{i} - \bar{y})^2 \tag{1.11}$$

Το R^2 είναι ένα στατιστικό μέγεθος το οποίο υπολογίζει ποιο ποσοστό της διασποράς του y (μεταβλητή απόκρισης) επεξηγείται από το χ (επεξηγηματική μεταβλητή). Το R^2 παίρνει τιμές από 0 εώς 1 και ισχύει ότι όσο πιο κόντα στο 1 είναι τόσο καλύτερα η μεταβλητή χ επεξηγεί την διασπορά της μεταβλητής y. Αν

το R^2 = 0.95 σημαίνει ότι η μεταβλητή χ επεξηγεί την διασπορά της μεταβλητής y κατά 95%. Δεν μπορεί να αποδοθεί μια συγκεκριμένη τιμή R^2 ώστε να ξέρουμε αν το μοντέλο μας είναι το κατάλληλο. Κατά την διάρκεια των χρόνων έχουν δοθεί διάφορες ιδέες για το ποια τιμή του R^2 είναι κατάλληλη. Μια απο αυτές είναι ότι αν R^2είναι κοντά στο 0.670 ή μεγαλύτερο το R^2 θεωρείται σημαντικό, αν R^2είναι κοντά στο 0,333 είναι μέτριο και αν R^2 είναι κοντά στο 0.190 ή μικρότερο θεωρείται μη σημαντικό ή αδύναμο (Chin,1998). Οι Hair et al. (2011) έθεσαν αυτά τα όρια στο 0.75, 0.5, 0.25. Ωστόσο ακόμα και αυτές οι ιδέες δεν πρέπει να λαμβάνονται ως αυθεντία και θα πρέπει να λαμβάνουμε υπόψιν από που προήλθαν τα δεδομένα μας. Κάποιες φορές ακόμα και R^2 κοντά στο 0.1 είναι ικανοποιητικό αν τα δεδομένα μας προέρχονται από κάποιον χώρο όπως το marketing (James et al.,2021). Τέλος στο απλό γραμμικό μοντέλο ο συντελεστής προσδιορισμού ταυτίζεται με το τετράγωνο του συντελεστή συσχέτισης Pearson ο οποίος δίνεται από την σχέση

$$r_{xy} = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(\mathrm{y_i}-\bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i-\bar{x})^2 \sum_{i=1}^{n}(\mathrm{y_i}-\bar{y})^2}} \quad (1.12)$$

ο οποίος μας δείχνει την γραμμική συσχέτιση των μεταβλητών x και y και όσο πιο κοντά είναι το R^2 στο 1 τόσο πιο γραμμική είναι η σχέση τους.

1.3 Συνθήκες Απλού Γραμμικού Μοντέλου

Προκειμένου να χρησιμοποιήσουμε το απλό γραμμικό μοντέλο θα πρέπει να λάβουμε υπόψιν και κάποιες αναγκαίες συνθήκες για την χρήση του. Οι συνθήκες αυτές είναι τέσσερις και είναι οι εξής:

- Γραμμικότητα: Στο απλό γραμμικό μοντέλο απαιτούμε η σχέση μεταξύ της επεξηγηματικής μεταβλητής και της μεταβλητής απόκρισης να είναι γραμμική. Αυτό μπορεί να εξεταστεί με διάφορους τρόπους ένας εκ των οποίων είναι η μελέτη του διαγράμματος διασποράς των τιμών των παρατηρήσεών μας $(\chi_i, \mathrm{y_i})$. Αν το διάγραμμα μας δεν είναι γραμμικό θα πρέπει είτε να μετασχηματίσουμε την μεταβλητή απόκρισης, είτε να χρησιμοποιήσουμε κάποιο άλλο μοντέλο. (Φουσκάκης, 2013)
- Κανονικότητα των σφαλμάτων: Θα πρέπει τα τυχαία σφάλματα που δίνονται από την σχέση (1.3) να ακολουθούν την κανονική κατανομή και ο έλεγχος αυτής της υπόθεσης μπορεί να γίνει είτε γραφικά ή με την χρήση

ενός τεστ Shapiro-Wilk (Hanusz et al., 2014). Στην περίπτωση που δεν ισχύει η κανονικότητα των σφαλμάτων θα πρέπει είτε να μετασχηματίσουμε την μεταβλητή απόκρισης είτε να χρησιμοποιήσουμε κάποιο άλλο μοντέλο.

- Ομοσκεδαστικότητα: Η υπόθεση της ομοσκεδαστικότητας υποδεικνύει ότι η διασπορά της μεταβλητής απόκρισης Y δοσμένης της τιμής χ της τυχαίας επεξηγηματικής μεταβλητής X παραμένει σταθερή ανεξάρτητα της τιμής χ, μαθηματικά δηλαδή $V(Y|X)$ $\sigma\tau\alpha\theta\varepsilon\rho\acute{\eta}$ $\forall\, x \in X$. Αυτό ισοδυναμεί με το ότι η διασπορά των τυχαίων σφαλμάτων ε_i είναι σταθερή $\forall\, x \in X$. Ο έλεγχος αυτής της υπόθεσης μπορεί να γίνει με την μελέτη του διαγράμματος διασποράς μεταξύ των υπολοίπων e_i και των προβλεπόμενων τιμών της μεταβλητής απόκρισης $\hat{y}_i$. Στην περίπτωση που τα ζεύγη αυτών των τιμών ακολουθούν κάποιο μοτίβο και δεν είναι τυχαία έχουμε το φαινόμενο της εταιροσκεδαστικότητας, το οποίο είναι ουσιαστικά η παραβίαση της ομοσκεδαστικότητας. Όταν συμβαίνει αυτό θα πρέπει είτε να μετασχηματίσουμε την μεταβλητή απόκρισης μας ώστε να ισχύει η ομοσκεδαστικότητα είτε να χρησιμοποιήσουμε κάποιο άλλο μοντέλο (Φουσκάκης, 2013)
- Ανεξαρτησία σφαλμάτων: Πρέπει τα τυχαία σφάλματα ε_i να είναι ανεξάρτητες τυχαίες μεταβλητές. Ο έλεγχος αυτής της υπόθεσης είναι δύσκολος και πολλές φορές δεν εφαρμόζουμε κάποιον έλεγχο καθώς μπορούμε να το καταλάβουμε από την φύση των μεταβλητών του πειράματός μας.

1.4 Μοντέλο Πολλαπλής Γραμμικής Παλινδρόμησης

Το μοντέλο της πολλαπλής γραμμικής παλινδρόμησης είναι μια προέκταση του απλού γραμμικού μοντέλου. Ουσιαστικά αντί να έχουμε ένα μοντέλο μιας μεταβλητής απόκρισης y και μιας μεταβλητής επεξήγησης x, όπως είχαμε στο απλό γραμμικό μοντέλο, έχουμε ένα μοντέλο μιας μεταβλητής απόκρισης y και πολλών μεταβλητών επεξήγησης $X_1, X_2, X_3, \dots, X_p$ με $p \in N$. Το μοντέλο αυτό το συμβολίζουμε ως

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \beta_3 x_{i3} + \dots + \beta_p x_{ip} + \varepsilon_i \qquad (1.13)$$

όπως και στο απλό γραμμικό μοντέλο τα $\beta_0, \beta_1, \beta_2, \dots, \beta_p$ με p ∈ N είναι οι συντελεστές της παλινδρόμησης και τα το ε_i συμβολίζει τα τυχαία σφάλματα. Όπως και στο απλό γραμμικό μοντέλο τα $\beta_0, \beta_1, \beta_2, \dots, \beta_p$ είναι άγνωστα και πρέπει να εκτιμηθούν και η μέθοδος με την οποία εκτιμούμε τις μεταβλητές αυτές είναι η μέθοδος των Ελάχιστων Τετραγώνων. Αφού εκτιμήσουμε τις μεταβλητές θα πάρουμε την σχέση

$$\widehat{y_i} = \widehat{\beta_0} + \widehat{\beta_1} x_{i1} + \widehat{\beta_2} x_{i2} + \widehat{\beta_3} x_{i3} + \dots + \widehat{\beta_p} x_{ip} \qquad (1.14)$$

Ο υπολογισμός αυτών των μεταβλητών γίνεται με στόχο την ελαχιστοποίηση της ποσότητας RSS που δίνεται από την σχέση (1.4) και αντικαθιστώντας την σχέση (1.14) σε αυτήν. Με την μέθοδο των Ελαχίστων Τετραγώνων λοιπόν υπολογίζουμε το διάνυσμα των $\hat{\beta} = (\widehat{\beta_0}, \widehat{\beta_1}, \widehat{\beta_2}, \dots, \widehat{\beta_p})$ από την σχέση (Καρώνη & Οικονόμου, 2017)

$$\hat{\beta} = (X'X)^{-1}X'y \qquad (1.15)$$

όπου X ο πίνακας σχεδιασμού και έχει την μορφή

$$X = \begin{array}{|c|c|c|c|c|} \hline 1 & x_{11} & x_{12} & \dots & x_{1p} \\ \hline 1 & x_{21} & x_{22} & \dots & x_{2p} \\ \hline \dots & \dots & \dots & \dots & \dots \\ \hline \dots. & \dots & \dots & \dots & \dots \\ \hline 1 & x_{n1} & x_{n2} & \dots & x_{np} \\ \hline \end{array}$$

X’ ο αντίστροφός του και ο y το διάνυσμα $y=(y_1, y_2, y_3, \dots y_n)'$. Και από την σχέση (1.15) καταλήγουμε στην σχέση

$$\hat{y} = X\widehat{\beta} \qquad (1.16)$$

που άμα την ανάξουμε μας δίνει την σχέση (1.14).

Οι συνθήκες οι οποίες πρέπει να ισχύουν για να μπορούμε να χρησιμοποιήσουμε το πολλαπλό γραμμικό μοντέλο ή για να είμαστε πιο ακριβείς για να έχει νόημα να χρησιμοποιήσουμε αυτό το μοντέλο, είναι οι ίδιοι με αυτούς του απλού γραμμικού μοντέλου. Δηλαδή:

- Γραμμικότητα
- Κανονικότητα σφαλμάτων
- Ομοσκεδαστικότητα
- Ανεξαρτησία σφαλμάτων

Επίσης όπως και στο απλό γραμμικό μοντέλο για να επιβεβαιώσουμε την ακρίβεια των συντελεστών του μοντέλου μας $\widehat{\beta_0}, \widehat{\beta_1}, \widehat{\beta_2}, \dots, \widehat{\beta_p}$ πρέπει να κάνουμε τους ελέγχους υποθέσεων (t-tests) σε 100(1-α)% διάστημα εμπιστοσύνης με αρχική υπόθεση ότι ο συντελεστής είναι 0 και ενναλακτική ότι είναι διάφορος του 0. Μαθηματικά οι έλεγχοι αυτοί γράφονται ως εξής:

- $H_0: \widehat{\beta_0} = 0$
  $H_1: \widehat{\beta_0} \neq 0$

- $H_0: \widehat{\beta_1} = 0$
  $H_1: \widehat{\beta_1} \neq 0$

(1.17)

- $H_0: \widehat{\beta_2} = 0$
  $H_1: \widehat{\beta_2} \neq 0 \dots$
  ...
  ...
- $H_0: \widehat{\beta_p} = 0$
  $H_1: \widehat{\beta_p} \neq 0$

Και απαιτούμε ο κάθε έλεγχος ξεχωριστά να μας δώσει p-value < α
Ο παραπάνω έλεγχος μπορεί να πραγματοποιηθεί επειδή οι συντελεστές

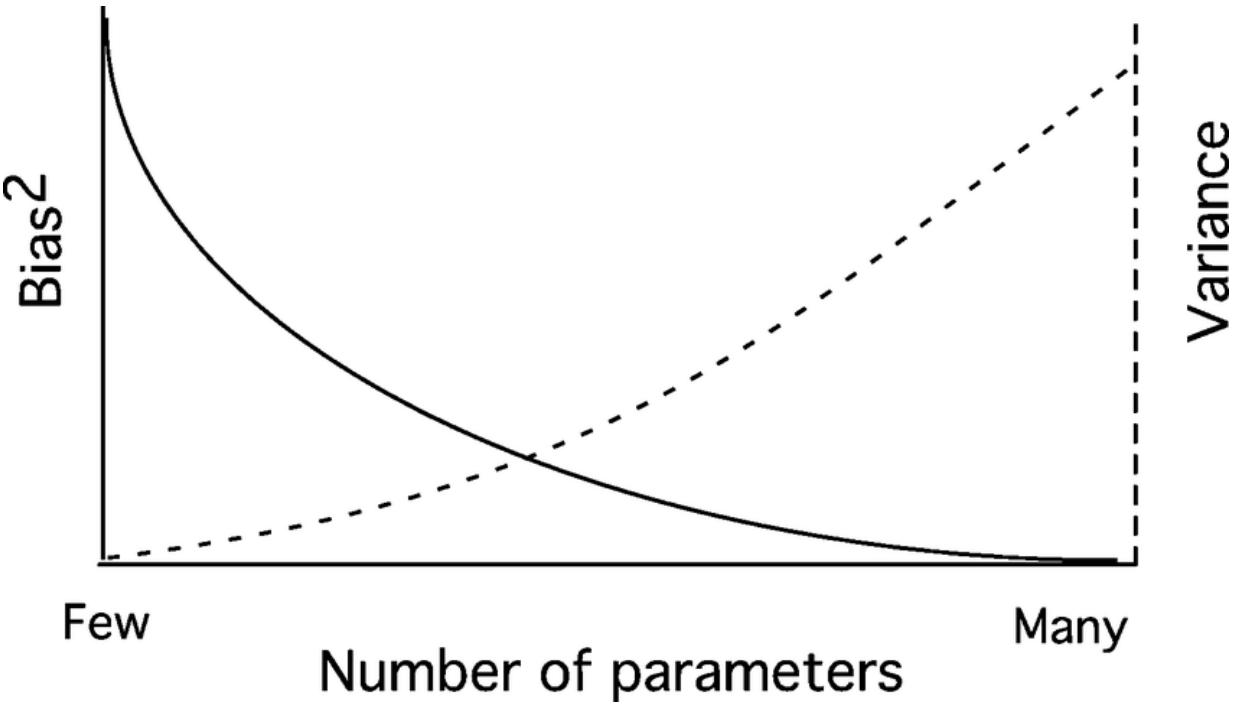
ακολουθούν την κανονική κατανομή. Γνωρίζουμε από την κανονικότητα των σφαλμάτων που ισχύει στο μοντέλο της πολλαπλής γραμμικής παλινδρόμησης ότι τα σφάλματα ακολουθούν την κανονική κατανομή. Επειδή το μοντέλο μας δίνεται από την σχέση (1.16) άμα θεωρήσουμε το X σταθερό εύκολα παρατηρούμε ότι και οι συντελεστές $\hat{\beta}$ ακολουθούν την κανονική κατανομή. Πριν πραγματοποιήσουμε τους ελέγχους (1.17) θα πραγματοποιήσουμε τον έλεγχο με αρχική υπόθεση ότι όλοι οι συντελεστές εκτός του $\widehat{\beta_0}$ είναι 0 και εναλλακτική ότι τουλάχιστον ένας είναι διάφορος του 0 ώστε να εξετάσουμε ότι το μοντέλο μας διαφέρει από το μοντέλο $\hat{y} = \widehat{\beta_0}$. Ο έλεγχος αυτός θα γίνει με την χρήση ένος ελέγχου F-test.

Τέλος στο πολλαπλό γραμμικό μοντέλο μπορούμε να ορίσουμε τις ποσότητες που ορίσαμε στο απλό γραμμικό μοντέλο RSS, RSE, TTS και τον συντελεστή R^2 που δίνονται από τις σχέσεις (1.4), (1.9), (1.11), (1.10) αντίστοιχα, ωστόσο δεν μπορούμε να εξετάσουμε την γραμμική συσχέτιση της μεταβλητής απόκρισης με τις επεξηγηματικές μεταβλητές από το πολλαπλό γραμμικό μοντέλο.

2 Μέθοδοι Επιλογής Μοντέλου και Μεταβλητών

2.1 Εισαγωγή

Πολλές φορές σε ένα σύνολο δεδομένων έχουμε μεγάλο αριθμό επεξηγηματικών μεταβλητών που καθιστά την πολλαπλή γραμμική παλινδρόμηση δυσνόητη και σύνθετη. Ο σκοπός μας σε κάθε στατιστικό μοντέλο που κατασκευάζουμε είναι να είναι κατανοητό και αποτελεσματικό. Αυτό το επιτυγχάνουμε ακολουθώντας την αρχή της οικονομίας (principle of parsimony) ή αλλιώς Occam's razor και έχει χρήσεις σε πολλές επιστήμες και όχι μόνο σε αυτήν της στατιστικής. Ο ορισμός αυτής της αρχής μπορεί να δοθεί ως εξής: «Όταν έχουμε δύο λύσεις εξίσου αποτελεσματικές, η πιο απλή είναι και η καλύτερη». Στην στατιστική η χρήση αυτή της αρχής μπορεί να περιγραφεί συνοπτικά από το Γράφημα 2.1.



Γράφημα 2.1: Αρχή της Οικονομίας (Occam's razor)

Ουσιαστικά στα μοντέλα μας αναζητούμε αυτό που είναι το πιο κατάλληλο (έχει χαμηλό bias) χρησιμοποιώντας όσο λιγότερες μεταβλητές μπορούμε.

Πολλές φορές η χρήση πολλών μεταβλητών για το μοντέλο μας οδηγεί στο πρόβλημα της υπερπροσαρμογής. Το πρόβλημα της υπερπροσαρμογής του μοντέλου δημιουργείται όταν το μοντέλο μας είναι υπερβολικά σύνθετο και λαμβάνει υπόψιν του ασήμαντα στοιχεία ενός συγκεκριμένου συνόλου δεδομένων και δεν συνεπάγεται στην γενίκευση του σε άλλα σύνολα δεδομένων. Η αρχή της οικονομίας μπορεί να περιγραφεί από την σχέση «σήματος» και «θορύβου» όπου στόχος ενός μοντέλου είναι να καταγράψει όλο το σήμα χωρίς να υπάρχει θόρυβος (Silver, 2012). Οι μεταβλητές οι οποίες δεν προσφέρουν κάτι ουσιαστικό στο μοντέλο μας ονομάζονται μη στατιστικά σημαντικές.

Για να αποφανθούμε ποιο είναι το καταλληλότερο μοντέλο για ένα σύνολο δεδομένων δηλαδή αυτό που περιγράφει καλύτερα τα δεδομένα μας κάποια από τα πιο γνωστά κριτήρια που χρησιμοποιούνται είναι:

- Ο συντελεστής προσδιορισμού R^2
- Ο διορθωμένος συντελεστής προσδιορισμού R^2_{adj}
- Στατική συνάρτηση C_p-Mallows
- AIC (Akaike Information Criterion)
- BIC (Bayesian Information Criterion)

Ενώ για την επιλογή μεταβλητών του μοντέλου μας κάποιες από τις πιο γνωστές μέθοδούς που χρησιμοποιούνται είναι οι εξής:

- Η μέθοδος της προς τα εμπρός επιλογής
- Η μέθοδος της προς τα πίσω απαλοιφής
- Η μέθοδος της κατά βήματα παλινδρόμησης .
- Παλινδρόμηση Lasso
- Αλγόριθμος παλινδρόμησης τυχαίων δασών (Random Forests)

Οι μέθοδοι για τα κριτήρια καταλληλότητας και τις μεθόδους επιλογής μεταβλητών για το μοντέλο μας θα εξηγηθούν εκτενώς στις ενότητες 2.2 και 2.3

2.2 Κριτήρια Καταλληλότητας

Προκειμένου να επιλέξουμε ποιο είναι το καταλληλότερο μοντέλο για ένα σύνολο δεδομένων χρησιμοποιούμε κάποια κριτήρια καταλληλότητας που αναφέραμε στην Παράγραφο 2.1. Είναι σημαντικό να σημειώσουμε ότι για να εφαρμόσουμε αυτά τα κριτήρια θα πρέπει να συγκρίνουμε μοντέλα που έχουν προσαρμοστεί στο ίδιο σύνολο δεδομένων.

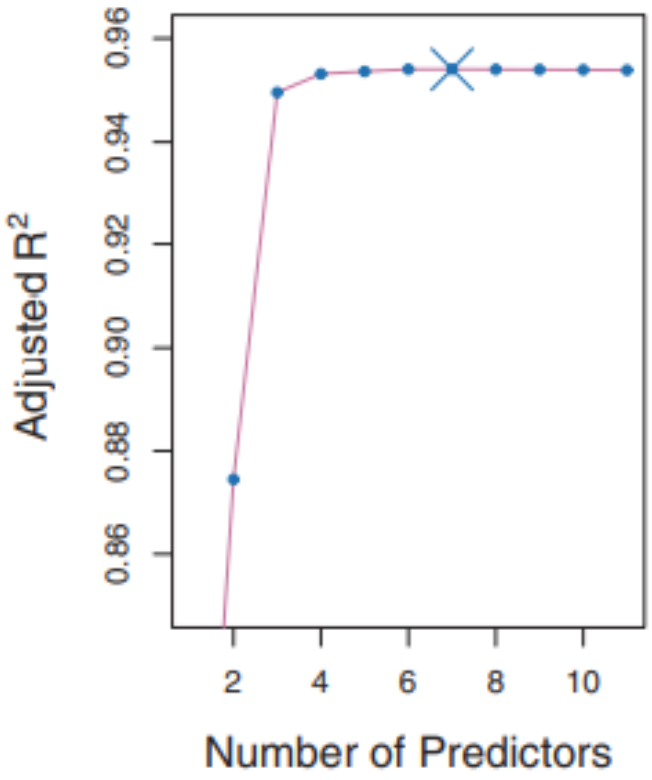
2.2.1 Ο Συντελεστής Προσδιορισμού R^2 και ο Διορθωμένος Συντελεστής Προσδιορισμού R^2_{adj}

Όπως αναφέραμε στην ενότητα 1.2 ο συντελεστής προσδιορισμού μας δείχνει ποιο ποσοστό της διασποράς της μεταβλητής απόκρισης εξηγείται από την επεξηγηματική μεταβλητή. Άμα προσθέσουμε μεταβλητές στο σύνολο δεδομένων μας, ο συντελεστής προσδιορισμού θα αυξηθεί ακόμα και αν αυτές οι μεταβλητές δεν είναι στατιστικά σημαντικές. Ένα αποτέλεσμα που δεν είναι επιθυμητό καθώς όπως αναφέρθηκε στην Παράγραφο 2.1 μπορεί να αποφέρει το πρόβλημα της υπερπροσαρμογής στο μοντέλο μας. Προκειμένου λοιπόν να χρησιμοποιήσουμε τον συντελεστή προσδιορισμού στο πολλαπλό γραμμικό μοντέλο θα πρέπει να του κάνουμε μερικές «διορθώσεις». Οπότε αντί να χρησιμοποιούμε το R^2 , χρησιμοποιούμε τον διορθωμένο συντελεστή προσδιορισμού R^2_{adj} που δίνεται από την σχέση

$$R^2_{adj} = 1 - \frac{\frac{RSS}{n-p-1}}{\frac{TSS}{n-1}} \qquad (2.1)$$

Όπου το RSS είναι η ποσότητα που έχει οριστεί στην σχέση (1.4), το TSS η ποσότητα που έχει οριστεί στην σχέση (1.11), το n είναι το πλήθος των παρατηρήσεων του σύνολο δεδομένων μας και το p είναι το πλήθος των μεταβλητών του μοντέλου. Η τιμή του διορθωμένου συντελεστή αυξάνεται όταν ο έλεγχος F για την πρόσθεση της μεταβλητής ξεπερνάει την μονάδα. (Καρώνη & Οικονόμου, 2017). Αυτό συμβαίνει διότι ο στόχος του διορθωμένου συντελεστή προσδιορισμού είναι η ελαχιστοποίηση της ποσότητας $\frac{RSS}{n-p-1}$ που όπως βλέπουμε λαμβάνει υπόψιν και τις μεταβλητές του μοντέλου. Στο Γράφημα 2.2 μπορούμε να δούμε ένα παράδειγμα όπου σε ένα μοντέλο με πολλές επεξηγηματικές μεταβλητές

ο R^2_{adj} δεν αλλάζει παρά την προσθήκη περισσότερων μεταβλητών και φαίνεται ότι ο ιδανικός αριθμός μεταβλητών που περιγράφει το μοντέλο μας από την αρχή της οικονομίας είναι οι 7 μεταβλητές



Γράφημα 2.2: Διορθωμένος συντελεστής προσδιορισμού συναρτήσει τον αριθμό των μεταβλητών ενός σύνολο δεδομένων. (*James et al., 2021, σ.211)*

2.2.2 AIC (Akaike Information Criterion)

Το κριτήριο AIC είναι ένα κριτήριο καταλληλότητας μοντέλου και δίνεται από την σχέση

$$AIC = 2d - 2lnL \tag{2.2}$$

Όπου d το πλήθος των παραμέτρων του μοντέλου και L η μεγιστοποιημένη τιμή της συνάρτησης πιθανοφάνειας για το μοντέλο που εκτιμάμε. (Akaike,1974) Προκειμένου να διαλέξουμε ένα μοντέλο βάσει της τιμής AIC του, επιλέγουμε αυτό που έχει την μικρότερη τιμή. Είναι σημαντικό να σημειώσουμε ότι αναφερόμαστε πάντα σε σύγκριση μοντέλων που προσαρμόζονται στο ίδιο σύνολο δεδομένων. Από τον τύπο του AIC καταλαβαίνουμε ότι το 2d δρα ως ποινή στο μοντέλο με στόχο να ποινικοποιεί την προσθήκη μεταβλητών στο μοντέλο μας οι οποίες δεν είναι στατιστικά σημαντικές.

Αξίζει να σημειωθεί ότι έχει αναπτυχθεί μια αναβάθμιση του κριτηρίου AIC, το διορθωμένο AIC ή αλλιώς Second Order Information Criterion που συμβολίζεται ως AIC_c (AIC corrected) και δίνεται από την σχέση

$$AIC_c = \mathrm{n}\left[\ln\left(\frac{2\pi SSR}{n}\right) + 1\right] + 2(p+1)\frac{n}{n-p-2} \tag{2.3}$$

Με p = k+1, k επεξηγηματικές μεταβλητές και n το μέγεθος του δείγματος μας. (Καρώνη & Οικονόμου, 2017)

Ένας ισοδύναμος ορισμός της σχέσης αυτής που μας δείχνει την συσχέτιση με το AIC είναι η εξής σχέση (Burnham & Anderson,2002)

$$AIC_c = AIC + \frac{2d\,(d+1)}{n-d-1} \tag{2.4}$$

Από τον ορισμό του διορθωμένου AIC μπορούμε να παρατηρήσουμε ότι όταν n->∞ το AIC_c = AIC. Συμπερασματικά μπορούμε να διαπιστώσουμε ότι σε μεγάλα δείγματα δεν έχει σημασία αν θα χρησιμοποιήσουμε το AIC ή το διορθωμένο AIC για την επιλογή του καταλληλότερου μοντέλου. Ενώ το διορθωμένο AIC πάντα είναι ένας καλύτερος «κριτής» από το AIC λόγω της δυσκολίας του στον υπολογισμό πολλές φορές προτιμούμε την χρήση του κριτηρίου AIC όταν μπορούμε να θεωρήσουμε ότι το αποτέλεσμα δεν θα έχει διαφορά. Ένα ακόμα κριτήριο που δόθηκε από τους Burnham και Anderson για να καταλάβουμε αν μπορούμε να χρησιμοποιήσουμε το κριτήριο AIC αντί του AIC_c, πέρα από το

πλήθος των παρατηρήσεων των δεδομένων μας, είναι ότι «το AIC_c πρέπει να χρησιμοποιηθεί εάν ο λόγος $\frac{n}{d} < 40$, δηλαδή μικρός». Ωστόσο πολλές φορές το γεγονός ότι έχουμε ένα αρκετά μεγάλο σύνολο δεδομένων επισκιάζει το σύνολο των μεταβλητών ακόμα και αν ο λόγος αυτός είναι μικρότερος από το 40.

2.2.3 BIC (Bayesian Information Criterion)

Όπως και το AIC, το BIC έχει ως στόχο την ποινικοποίηση της πρόσθεσης μη στατιστικά σημαντικών μεταβλητών. Το BIC δίνεται από τον τύπο

$$BIC = dln(n) - 2lnL \tag{2.5}$$

Όπου τα d, n, L ορίζουν τις ίδιες ποσότητες με αυτές που αναφέρθηκαν στην παράγραφο 2.2.2. Αξίζει να σημειωθεί ότι η βασική διαφορά του AIC και του BIC, είναι, ότι το BIC ψάχνει να βρει το «αληθινό» μοντέλο ανάμεσα σε αυτά που συγκρίνουμε ακόμα και αν αυτό το μοντέλο είναι χειρότερο από κάποιο άλλο το οποίο δεν έχουμε αναφέρει, ενώ το AIC προσπαθεί να επιλέξει το καταλληλότερο από τα μοντέλα που του δίνονται. Επίσης το BIC «τιμωρεί» περισσότερο την προσθήκη μεταβλητών. Γενικότερα το AIC προτιμάται σε προβλήματα με δεδομένα αντλημένα από τον πραγματικό κόσμο και σε προβλήματα όπου το μέγεθος του δείγματος είναι σχετικά μικρό.

2.3 Μέθοδοι Επιλογής Μοντέλου με Βήματα

Στην παράγραφό αυτήν θα ασχοληθούμε με τρεις μεθόδους που μπορούμε να χρησιμοποιήσουμε για να επιλέξουμε μοντέλο με βήματα και ουσιαστικά να επιλέξουμε τις κατάλληλες μεταβλητές για το μοντέλο μας που είναι οι εξής: η μέθοδος της προς τα εμπρός επιλογής, η μέθοδος της διαδοχικής αφαίρεσης και η μέθοδος της κατά βήματα εμπρός πίσω επιλογής. Αξίζει να σημειώσουμε ότι κάθε φορά που καταλήγουμε σε ένα μοντέλο χρησιμοποιώντας μια από αυτές τις μεθόδους θα πρέπει να εξετάζουμε και αν το μοντέλο που κατασκευάσαμε καλύπτει τις προϋποθέσεις του μοντέλου της πολλαπλής γραμμικής παλινδρόμησης που αναφέραμε στην Παράγραφο 1.4

2.3.1 Η Μέθοδος της Προς τα Εμπρός Επιλογής

Τα βήματα της μεθόδου της προς τα εμπρός επιλογής μπορούν να περιγραφούν ως εξής:

1. Προσαρμόζουμε το μοντέλο με έναν σταθερό όρο της μορφής $y=\beta_0$
2. Στην συνέχεια προσθέτουμε στο μοντέλο την μεταβλητή η οποία δίνει την μεγαλύτερη μείωση στο RSS, δηλαδή στην μεταβλητή που συμβάλλει περισσότερο στην επεξήγηση της μεταβλητής απόκρισης
3. Έπειτα προσαρμόζουμε το μοντέλο με την μεταβλητή του προηγούμενου βήματος και επαναλαμβάνουμε την διαδικασία αυτή για την επόμενη μεταβλητή που έχει την μεγαλύτερη επιρροή στο RSS
4. Επαναλαμβάνουμε τα δύο προηγούμενα βήματα μέχρι η προσθήκη κάποιας μεταβλητής να μην είναι στατιστικά σημαντική στο μοντέλο μας

Αξίζει να σημειώσουμε ότι με την μέθοδο αυτή ουσιαστικά ξεκινάμε με ένα μοντέλο που δεν επεξηγεί επαρκώς την μεταβλητή απόκρισης και προσθέτουμε μεταβλητές μέχρι να φτάσουμε στο σημείο οι μεταβλητές που προσθέτουμε να μην προσφέρουν κάποια στατιστικά σημαντική βελτίωση στην επεξήγηση της μεταβλητής απόκρισης.

2.3.2 Η Μέθοδος της Διαδοχικής Αφαίρεσης

Τα βήματα της μεθόδου της διαδοχικής αφαίρεσης είναι τα ακριβώς αντίθετα από αυτά της προς τα εμπρός επιλογής. Δηλαδή ξεκινάμε με όλες τις μεταβλητές στο μοντέλο μας και σιγά σιγά αφαιρούμε μέχρι να καταλήξουμε σε μια αφαίρεση μεταβλητής που επηρεάζει το μοντέλο μας σε στατιστικά σημαντικό βαθμό. Αναλυτικά:

1. Προσαρμόζουμε το μοντέλο μας με όλες τις μεταβλητές
2. Αφαιρούμε την λιγότερο σημαντική μεταβλητή και ελέγχουμε άμα αυτή η αλλαγή μεταβάλλει σημαντικά την διαφορά των RSS των δύο μοντέλων με έναν έλεγχο F
3. Προσαρμόζουμε το μοντέλο μας χωρίς την μεταβλητή του προηγούμενου βήματος και επαναλαμβάνουμε το βήμα 2 μέχρι η μεταβλητή που αφαιρούμε να επιφέρει στατιστικά σημαντική διαφορά μεταξύ των RSS των δύο μοντέλων

2.3.3 Η Μέθοδος της Κατά Βήματα Εμπρός Πίσω Επιλογής

Η μέθοδος αυτή ουσιαστικά είναι μια διόρθωση της μεθόδου της προς τα εμπρός επιλογής καθώς εξετάζει την περίπτωση που όταν προσθέτουμε μια μεταβλητή στο μοντέλο μας, η σημαντικότητα μιας άλλης μεταβλητής μικραίνει και κάνει έναν επιπλέον έλεγχο για να εξετάσει αν η μεταβλητή αυτή πρέπει να παραμείνει στο μοντέλο.

2.4 All Possible Regressions

Μια ακόμα μέθοδος που μπορεί να χρησιμοποιηθεί για να επιλέξουμε τις μεταβλητές που θα χρησιμοποιήσουμε στο μοντέλο μας είναι η All Possible Regressions Η μέθοδος All Possible Regressions προσαρμόζει την μεταβλητή απόκρισης χρησιμοποιώντας κάθε πιθανό συνδυασμό επεξηγηματικών μεταβλητών. Πρακτικά λαμβάνουμε τις τιμές AIC, BIC, Cp-Mallows , R^2 και άλλα στατιστικά για κάθε πιθανό μοντέλο που μπορούμε να κατασκευάσουμε από τα δεδομένα μας και χρησιμοποιώντας αυτά επιλέγουμε το καταλληλότερο. Η μέθοδος αυτή είναι αποτελεσματικότερη για την επιλογή των μεταβλητών από τις επιλογές με βήματα που αναλύσαμε στην Παράγραφο 2.3. Ωστόσο έχει ένα βασικό πρόβλημα, ότι όσες περισσότερες μεταβλητές έχει το μοντέλο μας τόσο πιο

χρονοβόρα και υπολογιστικά πολύπλοκη γίνεται. Για αυτόν τον λόγο δεν μπορεί να χρησιμοποιηθεί σε κάθε περίπτωση και αρκετές φορές προτιμούνται οι μέθοδοι της παλινδρόμησης κατά βήματα.

3 Μέθοδοι με την Χρήση Ποινής και ο Αλγόριθμος των Τυχαίων Δασών Παλινδρόμησης

3.1 Πολυσυγγραμικότητα

Το φαινόμενο της πολυσυγγραμικότητας είναι ένα φαινόμενο που συναντάμε όταν μια ή περισσότερες συμμεταβλητές του μοντέλου μας συνδέονται με μια γραμμική σχέση με μία ή περισσότερες από τις συμμεταβλητές. Το πρόβλημα που δημιουργείται λόγω αυτού του φαινομένου είναι το πόσο ευάλωτοι είναι οι συντελεστές σε αλλαγές άμα προστεθεί ή αφαιρεθεί κάποια μεταβλητή στο μοντέλο. Οπότε οι μέθοδοι της προς τα εμπρός επιλογής και της προς τα πίσω απαλοιφής που μελετήσαμε εκτενώς στις Παραγράφους 2.3.1 και 2.3.2 αντίστοιχα δεν μπορούν να χρησιμοποιηθούν ικανοποιητικά για την επιλογή μεταβλητών. Προκειμένου να εξετάσουμε αν οι συμμεταβλητές του συνόλου δεδομένων μας παρουσιάζουν το πρόβλημα της πολυσυγγραμικότητας υπάρχουν διάφοροι μέθοδοι που μπορούμε να εφαρμόσουμε ωστόσο αυτοί στους οποίους θα επικεντρωθούμε είναι ο γραφικός με τον οποίο μελετάμε μέσα από γραφικές παραστάσεις την γραμμική σχέση των συμμεταβλητών μεταξύ τους και το στατιστικό Παράγοντας Μεγέθυνσης Διασποράς (Variance Inflation Factor) γνωστό και ως VIF.

3.1.1 VIF (Variance Inflation Factor)

Η τιμή VIF ενός συντελεστή του μοντέλου δίνεται από την σχέση

$$VIF\left(\hat{\beta}_j\right) = \frac{1}{1-R^2_{X_j|X_{-j}}} \tag{3.1}$$

όπου το $R^2_{X_j|X_{-j}}$ είναι ο συντελεστής R^2 από την παλινδρόμηση της μεταβλητής X_j με τις υπόλοιπες (James et al., 2021) και όπως καταλαβαίνουμε όσο μεγαλύτερο το $R^2_{X_j|X_{-j}}$ τόσο πιο γραμμική η σχέση της μεταβλητής X_j με τις υπόλοιπες άρα και όσο μεγάλυτερη η τιμή VIF της συμμεταβλητής τόσο πιο γραμμική η σχέση της μεταβλητής X_j με τις υπόλοιπες. Οπότε όταν έχουμε μεγάλες τιμές VIF για αρκετές από τις συμμεταβλητές του μοντέλου μας μπορούμε εύκολα να διαπιστώσουμε ότι το σύνολο δεδομένων μας παρουσιάζει το φαινόμενο της πολυσυγγραμικότητας. Ωστόσο μια εύλογη απορία που προκύπτει από τα παραπάνω είναι η εξής: Τι θεωρούμε μεγάλη τιμή VIF για μια μεταβλητή;

Κατά καιρούς διάφοροι ερευνητές έχουν προσπαθήσει να θέσουν συγκεκριμένα όρια για να εξετάζεται η πολυσυγγραμικότητα του μοντέλου μέσω της χρήσης της τιμής VIF. Το πιο διαδεδομένο όριο που χρησιμοποιείται είναι ότι για τιμές VIF πάνω του 5 το σύνολο δεδομένων μας παρουσιάζει το φαινόμενο της πολυσυγγραμικότητας (James et al., 2021). Ωστόσο αξίζει να σημειωθούν και άλλοι περιορισμοί που έχουν δοθεί. Σύμφωνα με τους Hair et al. (2010) η μέγιστη επιτρεπτή τιμή VIF που πρέπει να έχει μια συμμεταβλητή είναι 10. Ενώ οι Kock & Lynn (2012) πρότειναν το όριο της τιμής VIF χαμηλότερα από το 5 στο 3.3. Γενικότερα το πόσο «ελαστικοί» είμαστε με τις τιμές VIF των συντελεστών έχει να κάνει με το σύνολο δεδομένων που μελετάμε. Στις περισσότερες περιπτώσεις μια τιμή VIF μεγαλύτερη από 5 είναι προβληματική για το σύνολο δεδομένων μας. Σε κάθε περίπτωση θα πρέπει να χρησιμοποιήσουμε και τον γραφικό έλεγχο περί συγγραμικότητας για να το επιβεβαιώσουμε.

3.1.2 Αντιμετώπιση της Πολυσυγγραμικότητας

Προκειμένου να αντιμετωπίσουμε το φαινόμενο της πολυσυγγραμικότητας οδηγούμαστε πέρα από την χρήση των μέθοδών που αναλύσαμε στο κεφάλαιο 2. Προτιμούμε να χρησιμοποιήσουμε μεθόδους παλινδρόμησης με ποινές ή κάποια μέθοδο μηχανικής μάθησης. Οι μέθοδοι ποινών που θα εξετάσουμε εκτενώς στο Κεφάλαιο 3 είναι η Παλινδρόμηση Κορυφογραμμής (Ridge Regression) και η παλινδρόμηση Lasso στις Παραγράφους 3.2 και 3.3 αντίστοιχα. Η μέθοδος μηχανικής μάθησης που θα αναλύσουμε είναι αυτή των δέντρων αποφάσεων πιο συγκεκριμένα των δέντρων παλινδρόμησης και των τυχαίων δασών (Random Forest) στο Κεφάλαιο 3.4.

3.2 Παλινδρόμηση Κορυφογραμμής (Ridge Regression)

Η παλινδρόμηση Ridge είναι μια τεχνική συρρίκνωσης και μοιάζει αρκετά με την μέθοδο των ελαχίστων τετραγώνων. Σε αντίθεση με αυτήν των ελαχίστων τετραγώνων που έχει ως στόχο την εκτίμηση των συντελεστών $\beta_0, \beta_1, \beta_2, \dots, \beta_p$ οι οποίες ελαχιστοποιούν την ποσότητα RSS η οποία δίνεται από την σχέση (1.4), η Ridge έχει ως στόχο την εκτίμηση συντελεστών $\hat{\beta}^R$οι οποιοί ελαχιστοποιούν την ποσότητα

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p}\beta_j^2 = RSS + \lambda \sum_{j=1}^{p}\beta_j^2 \quad (3.2)$$

Το λ ονομάζεται παράμετρος ποινής (tuning parameter) και επιλέγεται από αυτόν που κάνει την ανάλυση. Παρακάτω στην Παράγραφο 3.8 θα αναλύσουμε μεθόδους με τους οποίους μπορούμε να επιλέξουμε την καταλληλότερη τιμή λ για ένα στατιστικό μοντέλο. Ουσιαστικά αυτό που επιτυγχάνει η παράμετρος αυτή είναι ότι αν η ποσότητα $\lambda \sum_{j=1}^{p}\beta_j^2$ η οποία ονομάζεται shrinkage penalty είναι μικρή τότε οι συντελεστές β_j θα τείνουν στο 0. Παρατηρούμε επίσης ότι για λ=0 λαμβάνουμε την γραμμική παλινδρόμηση με την μέθοδο των ελαχίστων τετραγώνων και ότι όσο μεγαλώνει η τιμή του λ οι συντελεστές της παλινδρόμησης Ridge θα τείνουν στο 0 και έτσι θα καταλήξουμε σε ένα μοντέλο με μια σταθερά.

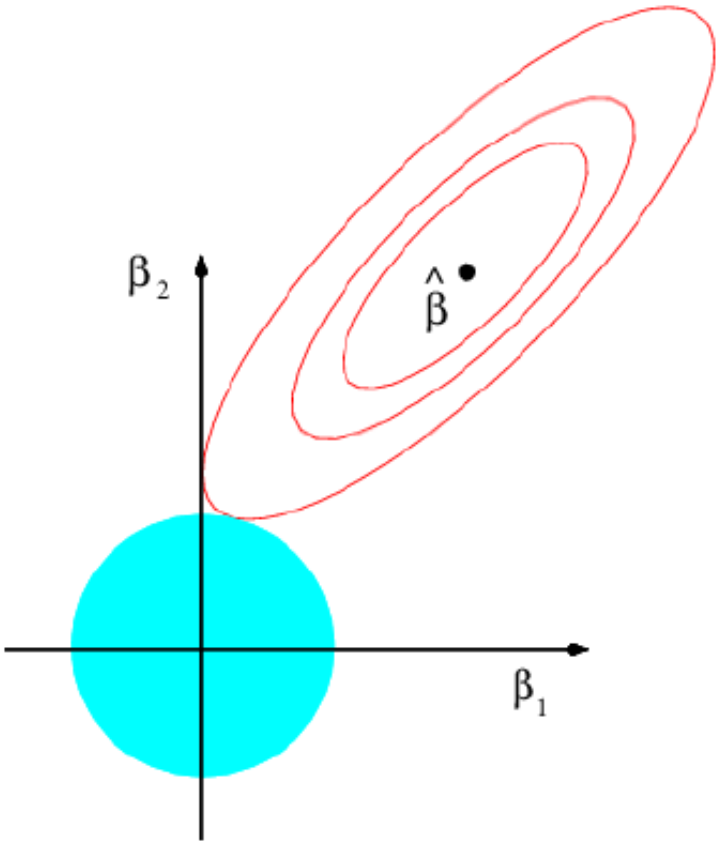
Ενδιαφέρον ωστόσο έχει ο λόγος για τον οποίον συμβαίνει αυτό. Η παλινδρόμηση Ridge μπορεί κανείς να πει ότι λύνει το εξής πρόβλημα:

Ελαχιστοποίηση της ποσότητας RSS για τις διάφορες τιμές του β με συνθήκη περιορισμού $\sum_{j=1}^{p}\beta_j^2 \leq s$ δηλαδή
$minimize_b(\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2)$ δεδομένου $\sum_{j=1}^{p}\beta_j^2 \leq s$.

Το πρόβλημα αυτό με την χρήση πολλαπλασιαστών Lagrange διαμορφώνεται ως η ελαχιστοποίηση του όρου

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda \sum_{j=1}^{p}\beta_j^2 \qquad (3.3)$$

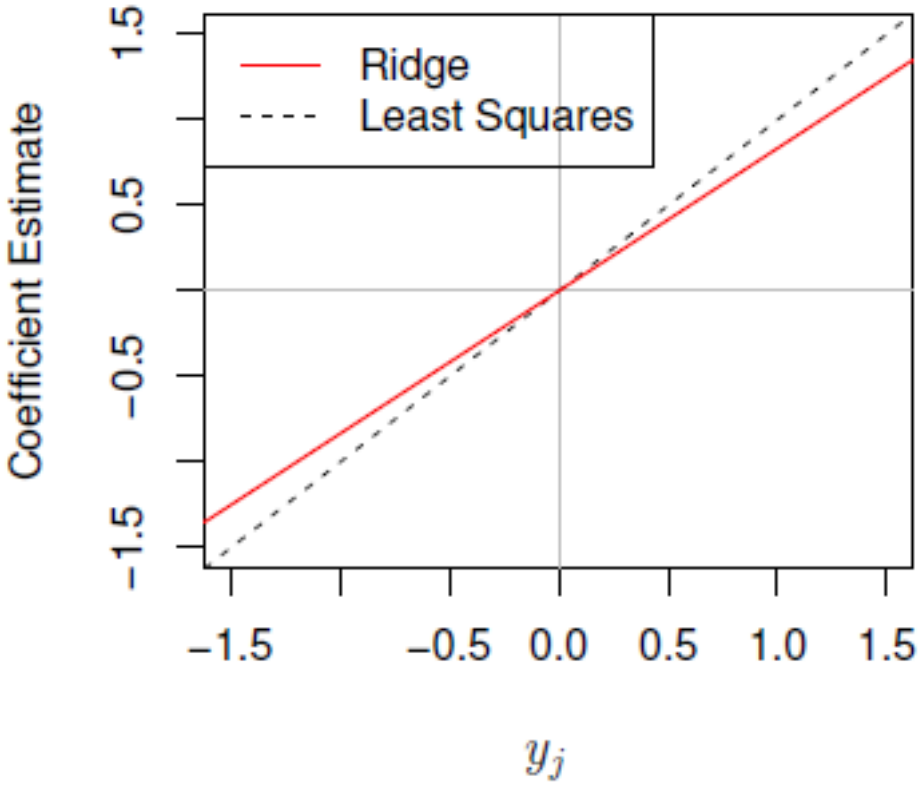
και έτσι καταλήγουμε στην ύπαρξη του λ. Το s υπάρχει ώστε να περιορίζει το μέγεθος της τιμής του $\sum_{j=1}^{p}\beta_j^2$. Για κάθε τιμή του λ λαμβάνουμε και το αντίστοιχο s και τα δύο μεγέθη αυτά είναι αντιστρόφως ανάλογα. Παρατηρούμε λοιπόν ότι για αρκετά μεγάλες τιμές του s η παλινδρόμηση Ridge θα μας δώσει τα ακριβώς ίδια αποτελέσματα με την μέθοδο των ελαχίστων τετραγώνων. Γραφικά μπορούμε να δούμε τις λύσεις της Ridge στο Γράφημα 3.1



Γράφημα 3.1: Γράφημα που περιγράφει το εύρος των τιμών των συντελεστών της παλινδρόμησης Ridge και των ελαχίστων τετραγώνων. (James et al.,2021, σ.244)

Στο Γράφημα 1 παρατηρούμε ότι η κόκκινη περιοχή είναι οι εκτιμήσεις για την μέθοδο των ελαχίστων τετραγώνων και η μπλε περιοχή είναι η $\beta_1^2 + \beta_2^2 \leq s$. Οι εκτιμητές βρίσκονται εκεί που η έλλειψη τέμνει τον κύκλο οπότε δεν μπορούν να είναι ποτέ 0.

Η παλινδρόμηση Ridge μας προσφέρει μια βελτιωμένη εκτίμηση των συντελεστών μας έναντι της μεθόδου των ελαχίστων τετραγώνων όταν στο δείγμα μας υπάρχει πολυσυγγραμικότητα. Την διαφορά μεταξύ των δύο αυτών μεθόδων μπορούμε να την δούμε γραφικά στους συντελεστές τους ως εξής:



Γράφημα 3.2: Γράφημα που συγκρίνει του συντελεστές στην παλινδρόμηση Ridge με αυτήν των ελάχιστων τετραγώνων στην περίπτωση μιας ανεξάρτητης μεταβλητής. (James et al.,2021, σ.248)

3.3 Παλινδρόμηση Lasso

Η παλινδρόμηση Lasso είναι μια μέθοδος παλινδρόμησης με ποινή όπως και η Ridge ωστόσο έχει ένα βασικό προτέρημα. Η Lasso μπορεί να χρησιμοποιηθεί ως μέθοδος επιλογής μεταβλητών, δηλαδή να επιλέξει τις κατάλληλες μεταβλητές για το μοντέλο μηδενίζοντας τους συντελεστές των μη σημαντικών μεταβλητών στην παλινδρόμηση.

Ο στόχος της Lasso είναι η εκτίμηση των συντελεστών $\hat{\beta}_\lambda^L$ που ελαχιστοποιούν την ποσότητα

$$\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2 + \lambda\sum_{j=1}^{p}|\beta_j| = \text{RSS} + \lambda\sum_{j=1}^{p}|\beta_j| \quad (3.4)$$

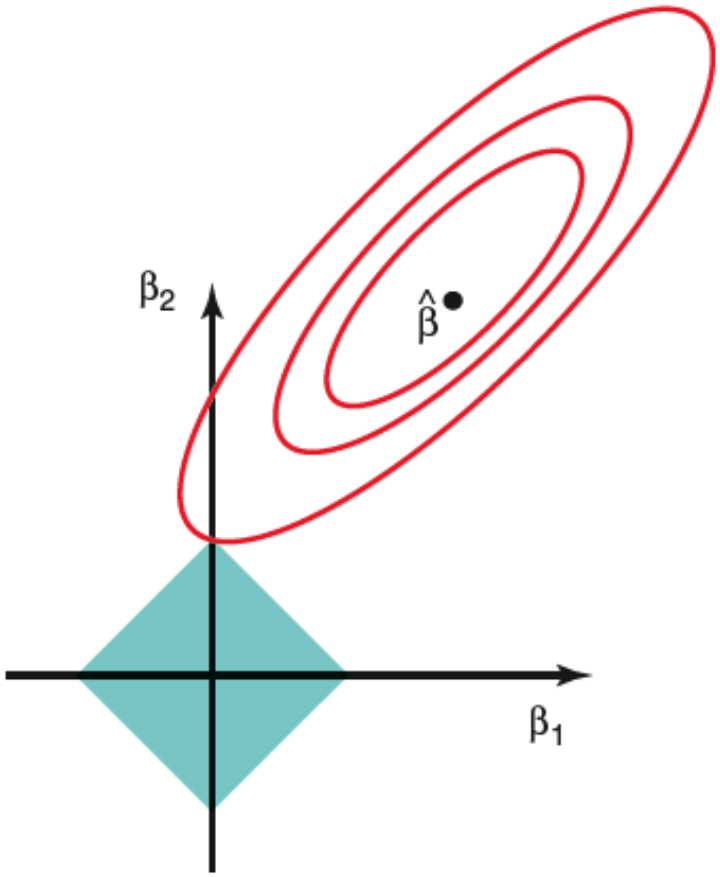
Παρατηρούμε ότι η βασική διαφορά με την παλινδρόμηση Ridge που επιδιώκει να εκτιμήσει τους συντελεστές που ελαχιστοποιούν την ποσότητα της σχέσης 3.2 βρίσκεται στο γεγονός ότι η Lasso χρησιμοποιεί σαν ποινή την L_1 ενώ η Ridge την νόρμα L_2. Να τονίσουμε ότι ποινή στην συγκεκριμένη ποσότητα της σχέσης 3.4 ονομάζεται ο παράγοντας $\lambda\sum_{j=1}^{p}|\beta_j|$. Αυτή η αλλαγή είναι και ο λόγος που η Lasso μπορεί να μηδενίσει τους συντελεστές μεταβλητών που είναι μη σημαντικές. Αξίζει επίσης να σημειώσουμε ότι όπως και στην Ridge η παράμετρος λ (tuning parameter) επιλέγεται με την χρήση της μεθόδου cross validation που θα μελετηθεί εκτενώς στην παράγραφο 3.5

Όπως και στην Ridge το πρόβλημα που προσπαθεί να λύσει η Lasso μπορεί να γραφτεί ως εξής:

Ελαχιστοποίηση της ποσότητας RSS για τις διάφορες τιμές του β με συνθήκη περιορισμού $\sum_{j=1}^{p}|\beta_j| \leq s$ δηλαδή
$minimize_\beta(\sum_{i=1}^{n}(y_i - \beta_0 - \sum_{j=1}^{p}\beta_j x_{ij})^2)$ δεδομένου $\sum_{j=1}^{p}|\beta_j| \leq s$.

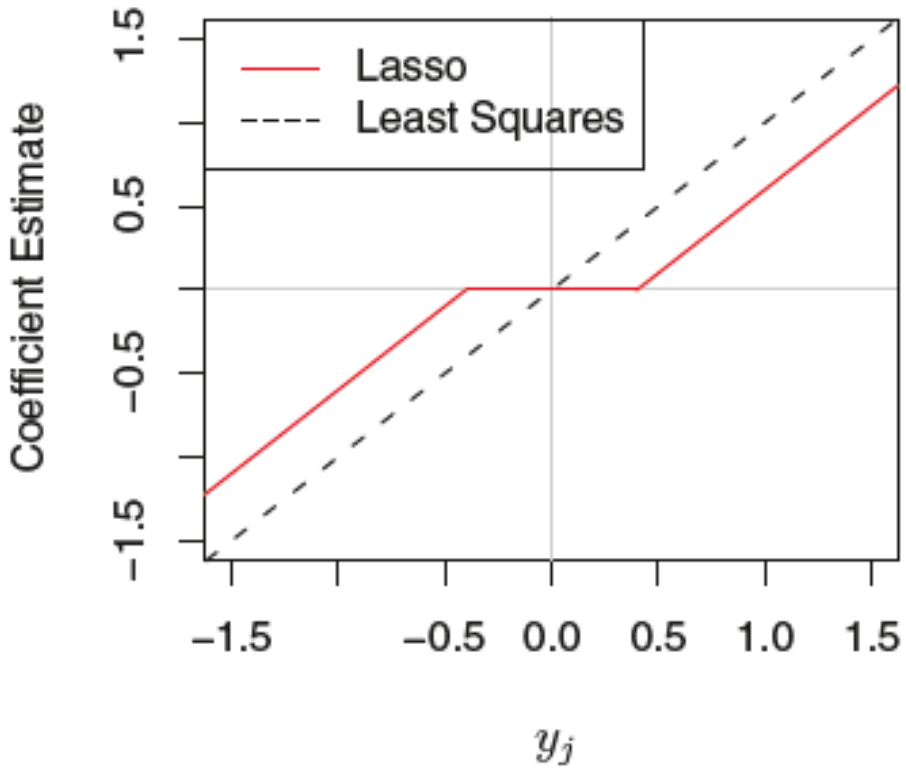
Ουσιαστικά είναι ένα πρόβλημα που μπορεί να λυθεί με την χρήση συντελεστών Lagrange. Όπως και στην παλινδρόμηση Ridge ο ρόλος που έχει η παράμετρος λ

είναι ο περιορισμός του αθροίσματος της απόλυτης τιμής των συντελεστών παλινδρόμησης για τις πολύ μεγάλες τιμές που μπορεί να λάβουν σε περίπτωση που το σύνολο δεδομένων μας «πάσχει» από το πρόβλημα της πολυσυγγραμικότητας. Επίσης όπως και στην παλινδρόμηση Ridge τα μεγέθη λ και s είναι αντιστρόφως ανάλογα. Η περιγραφή των λύσεων του προβλήματος, δηλαδή η εκτίμηση των συντελεστών μπορεί να δοθεί γραφικά από το Γράφημα 3.3



Γράφημα 3.3 Παρατηρούμε ότι η κόκκινη περιοχή είναι οι εκτιμήσεις για την μέθοδο των ελαχίστων τετραγώνων και η μπλε περιοχή είναι η $|\beta_1| + |\beta_2| \leq s$. Οι εκτιμητές βρίσκονται εκέι που η έλλειψη τέμνει τον κύκλο οπότε μπορούν να είναι 0. (James et al., 2021, σ.244)

Στο Γράφημα 3.3 παρατηρούμε ότι μπορεί να μηδενιστεί ένας συντελεστής Lasso σε αντίθεση με την παλινδρόμηση Ridge, ενώ από το γράφημα 3.4 παρατηρούμε την σχέση που έχει η Lasso με την μέθοδο των ελαχίστων τετραγώνων.

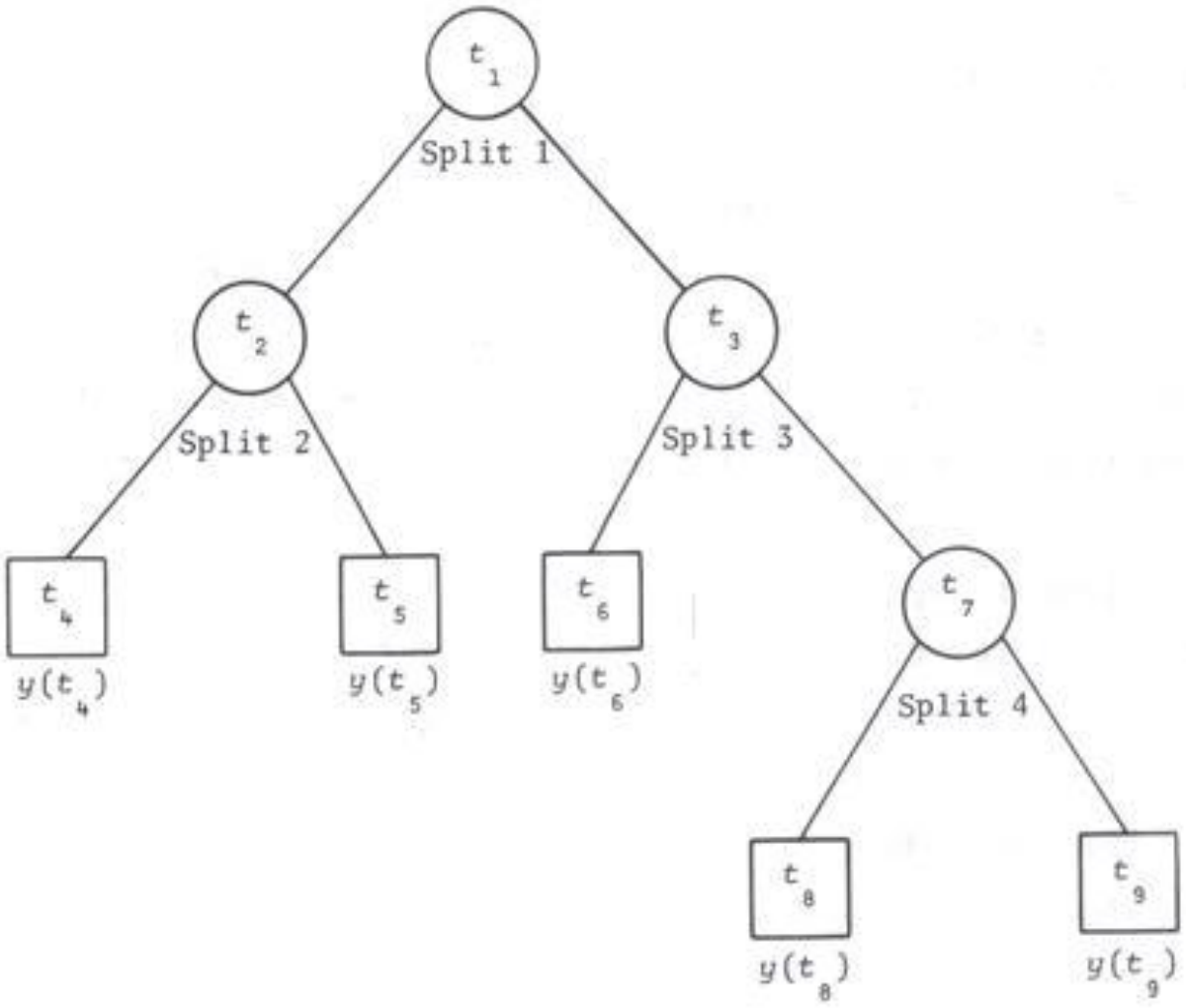


Γράφημα 3.4 Σύγκριση συντελεστών Lasso και συντελεστών με μέθοδο ελαχίστων τετραγώνων στην περίπτωση μιας ανεξάρτητης μεταβλητής. (James et al., 2021, σ.248)

3.4 Δέντρα Παλινδρόμησης (Regression Trees)

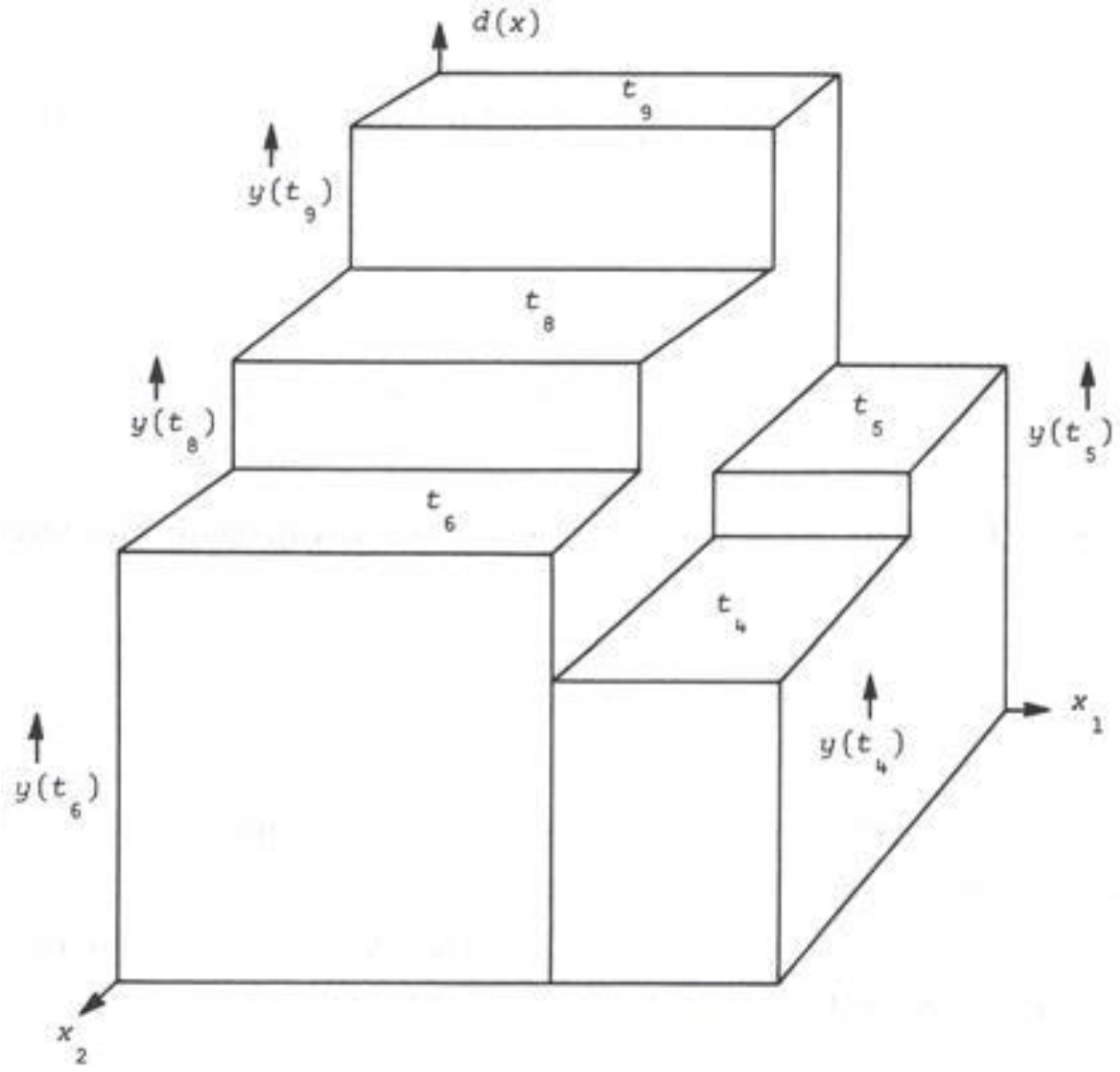
Τα δέντρα αποφάσεων (Decision Trees) αποτελούν ένα πολύ χρήσιμο εργαλείο για την μελέτη δεδομένων. Το μεγάλο προτέρημα τους απέναντι σε άλλες μεθόδους είναι η απλότητα τους και το πόσο εύκολο είναι να τα εξηγήσεις.

Τα δέντρα αποφάσεων χωρίζονται σε δύο κατηγορίες: στα δέντρα ταξινόμησης (Classification Trees) και στα δέντρα παλινδρόμησης (Regression Trees). Η διαφορά των δύο αυτών δέντρων αποφάσεων έχει να κάνει με το τι αναπαριστά η μεταβλητή απόκρισης στο πρόβλημά μας. Αν η μεταβλητή απόκρισης είναι κατηγορική τότε έχουμε ένα πρόβλημα ταξινόμησης και χρειαζόμαστε ένα δέντρο ταξινόμησης. Αν η μεταβλητή απόκρισης είναι ποσοτική μεταβλητή τότε για την επίλυση του προβλήματος χρειαζόμαστε ένα δέντρο παλινδρόμησης. Τα δέντρα αποφάσεων στα οποία θα επικεντρωθούμε είναι αυτά της παλινδρόμησης, τα οποία λειτουργούν ως εξής: Στόχος τους είναι η κατασκευή ενός δέντρου που θα προβλέπει με βάση τις μεταβλητές του συνόλου δεδομένων μας την μεταβλητή απόκρισης. Το δέντρο αποτελείται από κόμβους (nodes) και κλαδιά (branches) τα οποία και συνδέουν τους κόμβους. Οι κόμβοι περιέχουν την μεταβλητή την οποία εξετάζουμε. Κάθε κόμβος χωρίζεται σε δύο κλαδιά (συνήθως αλλά δεν θεωρείται απαραίτητο). Όταν ο κόμβος είναι μια κατηγορική μεταβλητή το ένα κλαδί είναι η επιλογή να ανήκει σε αυτήν την κατηγορία και το άλλο κλαδί να μην ανήκει σε αυτήν την κατηγορία. Ενώ στην περίπτωση που ο κόμβος είναι μια ποσοτική μεταβλητή, τότε το ένα κλαδί περιλαμβάνει τιμές της μεταβλητής του κόμβου που είναι μικρότερες από μια ορισμένη τιμή (θα επεξηγήσουμε στην συνέχεια πως προκύπτει) και το άλλο κλαδί τιμές μεγαλύτερες από αυτήν την τιμή. Στα άκρα των κλαδιών υπάρχουν πάλι κόμβοι οι οποίοι ονομάζονται εσωτερικοί κόμβοι (internal nodes). Η διαδικασία αυτή επαναλαμβάνεται μέχρι να καταλήξουμε σε μια τιμή στο τέλος ενός κλαδιού που αντί για κόμβο θα είναι μια τιμή πρόβλεψης της μεταβλητής απόκρισης για τις αντίστοιχες τιμές των μεταβλητών που έχουν προηγηθεί στα κλαδιά που συνδέουν τους κόμβους. Αυτή η τιμή αποτελεί τον τερματικό κόμβο του δέντρου.



Γράφημα 3.5 Γράφημα ενός παραδείγματος δέντρου παλινδρόμησης με 5 τερματικούς κόμβους (Breiman et al., 1984, σ.229)

Οι τιμές τις οποίες λαμβάνουν τα κλαδιά $t_1, t_2, \dots, t_9$ προκύπτουν από μια διαδικασία διάκρισης των δεδομένων μας σε περιοχές. Οι περιοχές που περιγράφουν το δέντρο του Γραφήματος 3.5 φαίνονται στο Γράφημα 3.6. Τις περιοχές αυτές ονομάζουμε R_j και ουσιαστικά ισχύει $R_j = y(t_j)$. Δηλαδή, οι περιοχές είναι οι τερματικοί κόμβοι του δέντρου.



Γράφημα 3.6 Γράφημα που περιγράφει τις περιοχές R_j . Στο συγκεκριμένο γράφημα αναφέρονται ως y(t_j). (Breiman et al.,1984, σ.230)

Η κατασκευή αυτών των περιοχών μπορεί να περιγραφεί από μια διαδικασία δύο βημάτων. Αρχικά χωρίζουμε τον χώρο μας με τις τιμές της μεταβλητής απόκρισης του συνόλου δεδομένων μας σε j διαφορετικές περιοχές $R_1, R_2, \dots, R_J$ οι οποίες έχουν σχήμα κουτιού και στην συνέχεια υπολογίζουμε την μέση τιμή των

παρατηρήσεων που ανήκουν στην κάθε περιοχή και έτσι βρίσκουμε τις τιμές $y(t_j)$ για κάθε περιοχή όπου j=1,2,…,J. Αυτό το κάνουμε για οποιοδήποτε σύνολο μεταβλητών ώστε να καταλήξουμε σε ένα δέντρο που περιέχει όλες τις μεταβλητές του συνόλου δεδομένων μας. Η επιλογή των περιοχών $R_1, R_2, \dots, R_J$ γίνεται έτσι ώστε να λάβουμε το μικρότερο δυνατό άθροισμα RSS των περιοχών που δίνεται από την σχέση (James et al., 2021)

$$\sum_{j=1}^{J} \sum_{i \in R_j} (y_i - \hat{y}_{R_j})^2 \tag{3.5}$$

Όπου το $\hat{y}_{R_j}$ είναι η μέση τιμή των παρατηρήσεων που ανήκουν στο κουτί j (ή αλλιώς στην περιοχή R_j).

Ένα ακόμα μέτρο που μπορεί να χρησιμοποιηθεί για την κατάλληλη επιλογή του δέντρου παλινδρόμησης είναι ένα στατιστικό που χρησιμοποιείται και για τα δέντρα ταξινόμησης το impurity function που δίνεται από την σχέση (Louppe,2014)

$$i_R(R_j) = \frac{1}{N_{R_j}} \sum (y_i - \hat{y}_{R_j})^2 \tag{3.6}$$

όπου N_{R_j} ο αριθμός των περιοχών R_j στο δέντρο μας.

Αξίζει να σημειωθεί ότι όταν έχουμε παραπάνω από δύο επεξηγηματικές μεταβλητές (δηλαδή η πιο σύνηθης περίπτωση) στο πρόβλημα μας, η διαδικασία που πρέπει να ακολουθήσουμε είναι η κατασκευή περιοχών $R_1(j,s) = \{X | X_j < s\}$ και $R_2(j,s) = \{X | X_j \geq s\}$ και να βρούμε τις τιμές j και s που ελαχιστοποιούν την ποσότητα (James et al., 2021)

$$\sum_{x_i \in R_1(j,s)} (y_i - \hat{y}_{R_1})^2 + \sum_{x_i \in R_2(j,s)} (y_i - \hat{y}_{R_2})^2 \tag{3.7}$$

Όπου το X_j είναι μια από τις μεταβλητές του προβλήματος. Συνεχίζουμε την ίδια διαδικασία χρησιμοποιώντας τις δύο περιοχές που έχουμε ήδη δημιουργήσει και τις διαχωρίζουμε και αυτές σε μικρότερες περιοχές. Η μέθοδος αυτή ονομάζεται «Binary Recursive Partitioning».

Ωστόσο τα δέντρα παλινδρόμησης όπως τα έχουμε περιγράψει έχουν ένα βασικό πρόβλημα. Λόγω των πολλών τερματικών κόμβων που έχουμε στο μοντέλο μας έχουμε εισάγει και επιπλέον θόρυβο, με αποτέλεσμα να «πάσχει» από το πρόβλημα της υπερπροσαρμογής (overfitting), όπως εξηγήσαμε στην Παράγραφο

2.1. Για να αντιμετωπίσουμε αυτό το πρόβλημα, στο δέντρο παλινδρόμησης που έχουμε κατασκευάσει, θα χρησιμοποιήσουμε την μέθοδο κλαδέματος (pruning). Στόχος μας είναι να μειώσουμε τους τερματικούς κόμβους (terminal nodes) με αποτέλεσμα την μείωση της διασποράς του δέντρου με πολύ χαμηλό κόστος στην αύξηση της μεροληψίας του (αρχή της οικονομίας). Στόχος της μεθόδου κλαδέματος είναι η ελαχιστοποίηση της σχέσης (James et al.,2021)

$$\sum_{m=1}^{|T|} \sum_{i:x_i \in R_m} (y_i - \hat{y}_{R_m})^2 + \alpha|T| \qquad (3.8)$$

Το |T| είναι ο αριθμός των τερματικών κόμβων του δέντρου και το R_m είναι οι περιοχές που αντιστοιχούν στον m τερματικό κόμβο. Αξίζει να σημειωθεί ότι η σχέση 3.8 θυμίζει την παλινδρόμηση Lasso και την παλινδρόμηση Ridge όπου το α αποτελεί μια ρυθμιστική παράμετρο (tuning parameter). Σε κάθε τιμή του α αντιστοιχεί ένα δέντρο το οποίο είναι υποσύνολο του αρχικού δέντρου που είχαμε κατασκευάσει. Αν το α=0 τότε το δέντρο που λαμβάνουμε είναι το αρχικό μας δέντρο. Επίσης αν το α είναι μικρό τότε το $|T|$ θα είναι μεγάλο (Breiman et al.,1984). Τέλος, η μέθοδος που χρησιμοποιούμε για να βρούμε την κατάλληλη τιμή του α είναι η ίδια που χρησιμοποιούμε για να βρούμε την ρυθμιστική παράμετρο στις Lasso και Ridge, η cross-validation. Η μέθοδος pruning είναι αποτελεσματικότερη από την χρήση κάποιου κριτηρίου το οποίο σταματάει το δέντρο πριν ολοκληρωθεί. Και αυτός είναι ο λόγος που δεν θα αναφερθούμε σε αυτά τα κριτήρια.

Το βασικό πρόβλημα που έχει η μέθοδος των δέντρων παλινδρόμησης, είναι ότι παρά την απλότητα του, υστερεί σε σημαντικό βαθμό στην ακρίβεια πρόβλεψης σε σχέση με άλλες μεθόδους. Ο τρόπος βελτίωσής της είναι η παραγωγή πολλών δέντρων με την χρήση των μεθόδων Bagging, Random Forests και Boosting που θα αναλυθούν εκτενώς στις Παραγράφους 3.5, 3.6, 3.7

3.5 Bootstrap Aggregation (Bagging)

Όπως αναφέραμε και στην παράγραφο 3.4 τα δέντρα αποφάσεων. Παρά το γεγονός ότι είναι απλά στην κατανόηση και την επεξήγηση, πάσχουν από το ότι δεν είναι ακριβή καθώς έχουν μεγάλη διασπορά. Το πρόβλημα αυτό έρχεται να λύσει η μέθοδος bagging. Η μέθοδος αυτή έχει ως στόχο την μείωση της διασποράς και αξίζει να σημειωθεί ότι η χρήση της δεν περιορίζεται μόνο στα δέντρα αποφάσεων, αλλά βρίσκει χρήσεις και σε άλλα μοντέλα στατιστικής εκμάθησης. Ουσιαστικά ο τρόπος που λειτουργεί είναι ο εξής: Προκειμένου να μειώσουμε την διασπορά και κατά συνέπεια να βελτιώσουμε την ακρίβεια του μοντέλου θα ήταν ιδανικό να πάρουμε διάφορα δεδομένα εκμάθησης (training sets), τα οποία τα χρησιμοποιούμε για να εκπαιδεύσουμε το μοντέλο μας και να βελτιώσουμε την ακρίβεια του για το σύνολο δεδομένων. Γενικά training set ονομάζουμε ένα υποσύνολο του συνόλου δεδομένων μας που χρησιμοποιούμε σε συνδυασμό με ένα άγνωστο σύνολο δεδομένων για να εκπαιδεύσουμε αυτό το άγνωστο σύνολο δεδομένων. Έστω Β τέτοια training sets θα κατασκευάσουμε Β διαφορετικά μοντέλα πρόβλεψης, κάτι το οποίο δεν είναι πρακτικό καθώς από ένα σύνολο δεδομένων παίρνουμε μόνο ένα training set. Οπότε χρησιμοποιούμε την μέθοδο boostrap με την οποία μετασχηματίζουμε το training set, που έχουμε, σε Β διαφορετικά training sets. Έτσι θα πάρουμε Β διαφορετικές προβλέψεις και υπολογίζοντας το μέσο τους θα πάρουμε μια πρόβλεψη για το αρχικό μοντέλο μας η οποία είναι μικρότερη σε διασπορά και πιο ακριβής από την αρχική μας πρόβλεψη. Αφού εκπαιδεύσουμε την μέθοδο μας στο Β-οστό training set λαμβάνουμε την πρόβλεψη $\hat{f}^{*b}(x)$ και υπολογίζοντας το μέσο με τον τύπο

$$\hat{f}_{bag}(x) = \frac{1}{B}\sum_{b=1}^{B}\hat{f}^{*b}(x) \tag{3.9}$$

Αυτόν τον τρόπο εφαρμόζουμε και στα δέντρα αποφάσεων. Παράγουμε Β δέντρα αποφάσεων και Β training sets που έχουν προέλθει από το ίδιο training set απλά μετασχηματίζοντας το. Τα δέντρα αυτά χρησιμοποιούν όλες τις μεταβλητές, καθώς δεν έχει χρησιμοποιηθεί σε αυτά η μέθοδος pruning, οπότε, έχουν πολύ υψηλή διασπορά και με την μέθοδο bagging την μειώνουμε σημαντικά. Έτσι καταλήγουμε σε ένα μοντέλο με χαμηλή μεροληψία και χαμηλή διασπορά. Ωστόσο η μέθοδος bagging δεν είναι η ιδανικότερη βελτίωση των δέντρων

αποφάσεων καθώς λαμβάνουν υπόψιν όλες τις μεταβλητές στα δέντρα τα οποία παράγουν. Η μέθοδος η οποία λύνει αυτό το πρόβλημα είναι ο αλγόριθμος των τυχαίων δασών (Random Forests) που θα αναλυθεί εκτενώς στην παράγραφο 3.

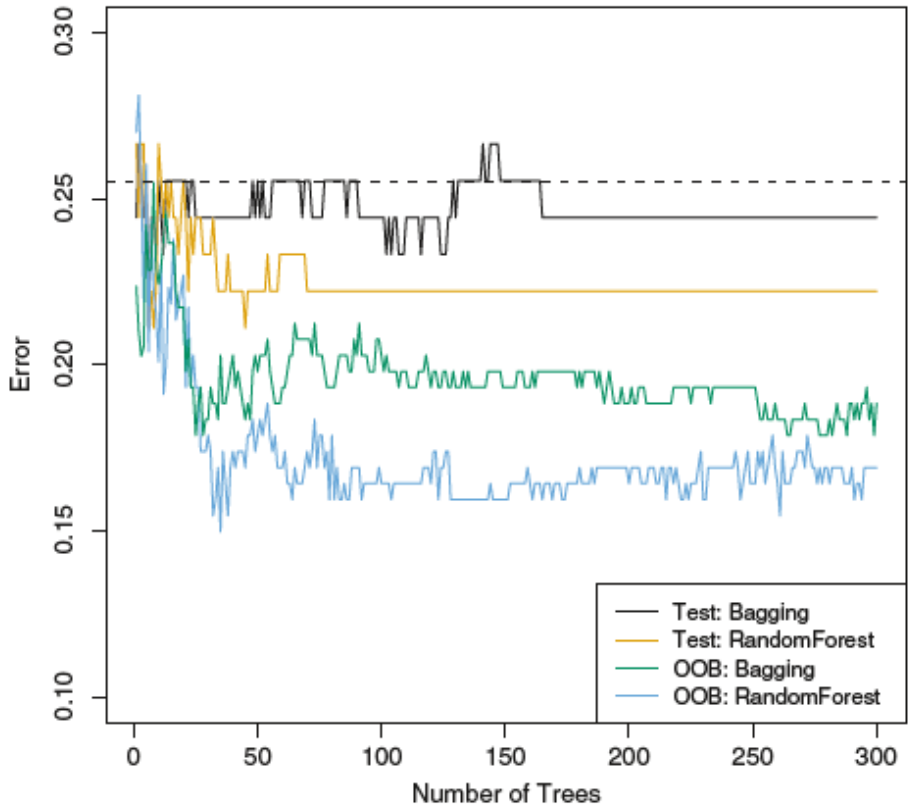
3.6 Αλγόριθμος των Τυχαίων Δασών (Random Forests)

Ο αλγόριθμος των τυχαίων δασών (Random Forests) αποτελεί μια βελτίωση της μεθόδου bagging στην σύνθεση πολλών δέντρων παλινδρόμησης. Η μέθοδος Random Forests, αντί να αφήνει τα δέντρα παλινδρόμησης να χρησιμοποιούν όλες τις μεταβλητές του προβλήματος και να μεγαλώνουν μέχρι τέλους, περιορίζει τα δέντρα να χρησιμοποιούν συγκεκριμένο αριθμό μεταβλητών. Αξίζει να σημειωθεί ότι αν κατασκευάσουμε ένα τυχαίο δάσος χρησιμοποιώντας ως αριθμό μεταβλητών όλες τις μεταβλητές του συνόλου δεδομένων μας θα λάβουμε ακριβώς το ίδιο αποτέλεσμα με την μέθοδο bagging.

Ο λόγος που παρατηρούμε βελτίωση στο μοντέλο μας χρησιμοποιώντας λιγότερες μεταβλητές για την κατασκευή των δέντρων, είναι ότι όταν χρησιμοποιούμε όλες τις μεταβλητές, τα δέντρα που κατασκευάζουμε χρησιμοποιούν σχεδόν πάντα την ίδια μεταβλητή, ως την πιο σημαντική. Ως αποτέλεσμα τα δέντρα που κατασκευάζονται με την μέθοδο bagging να μοιάζουν αρκετά μεταξύ τους, οπότε και να συσχετίζονται μεταξύ τους. Συμπερασματικά δεν παρατηρούμε μεγάλη διαφορά στο πρόβλημα της διασποράς που παρουσιάζουν τα δέντρα αποφάσεων. Εν αντιθέσει, όταν θέτουμε ένα όριο στο δέντρο σχετικά με τον αριθμό των μεταβλητών που θα χρησιμοποιήσει, κατασκευάζονται και δέντρα που δεν περιέχουν καθόλου την σημαντικότερη μεταβλητή. Το γεγονός αυτό έχει ως αποτέλεσμα να προσθέτουμε διασπορά στο πρόβλημα μας και να μην παρατηρείται έντονα το φαινόμενο της συσχέτισης.

Ο τρόπος που αξιολογούμε την καταλληλότητα του μοντέλου μας που έχει δημιουργηθεί είτε μέσω Bagging είτε με την μέθοδο Random Forests, είναι με την χρήση ενός στατιστικού που ονομάζεται Out Of Bag error estimate (OOB error), το οποίο είναι αρκετά ακριβές για να υπολογίζει το σφάλμα γενίκευσης του μοντέλου (Breiman, 1996). Το σφάλμα γενίκευσης, ουσιαστικά είναι το σφάλμα που θα έχει το μοντέλο που κατασκευάσαμε για την μεταβλητή απόκρισης, αν χρησιμοποιηθεί σε ένα άγνωστο σύνολο ελέγχου. Στην ιδανική περίπτωση, η διαδικασία, η οποία ακολουθείται για να εξεταστεί το σφάλμα γενίκευσης, είναι η συλλογή δεδομένων πέρα από το σύνολο δεδομένων που έχουμε στην διάθεση μας. Η συγκεκριμένη διαδικασία είναι σχετικά χρονοβόρα και σε πολλές

περιπτώσεις αδύνατη σε σχέση με την χρήση του OOB error, με αποτέλεσμα το OOB error να προτείνεται ως η καταλληλότερη μέθοδος. Η λογική πίσω από το OOB error είναι η εξής: χρησιμοποιούμε για την κατασκευή κάθε δέντρου τα 2/3 των παρατηρήσεων μας και στην συνέχεια εξετάζουμε πως λειτουργεί το μοντέλο, το οποίο κατασκευάσαμε χρησιμοποιώντας αυτές τις προβλέψεις στο υπόλοιπο 1/3 των παρατηρήσεων μας. Το δείγμα αυτό ονομάζεται Out Of Bag δείγμα (OOB sample). Σε προβλήματα παλινδρόμησης λαμβάνουμε το MSE του δείγματος, που δεν λάβαμε υπόψιν, όταν κατασκευάσαμε το μοντέλο μας. Χρησιμοποιούμε ως τιμή πρόβλεψης την τιμή που πήραμε από το μοντέλο μας και ως παρατηρήσεις τις τιμές των παρατηρήσεων που δεν λάβαμε υπόψιν. Αυτό ονομάζεται OOB MSE και όπως είπαμε και παραπάνω είναι ένα ικανοποιητικό στατιστικό για να εξετάσει το σφάλμα γενίκευσης. Αξίζει να σημειώσουμε ότι σε περιπτώσεις ταξινόμησης αντί για το OOB MSE λαμβάνουμε υπόψιν το σφάλμα ταξινόμησης.



Γράφημα 3.7 Γράφημα το οποίο εξετάζει το Test και το OOB error μεταξύ της μεθόδου Random Forests και Bagging σε ένα συγκεκριμένο σύνολο δεδομένων. (James et al., 2021, σ.318)

Παρατηρούμε από το Γράφημα 3.7 ότι όσο πιο πολύ αυξάνεται ο αριθμός των δέντρων τόσο καλύτερα λειτουργεί ο αλγόριθμος των τυχαίων δασών εναντίον της μεθόδου Bagging, κάτι το οποίο ήταν αναμενόμενο να παρατηρήσουμε βάσει όσων προαναφέραμε.

Αξίζει να σημειωθεί ότι ο αριθμός ο οποίος είναι η προκαθορισμένη επιλογή στις βιβλιοθήκες της R και της Python που εκτελούν την μέθοδο Random Forests για προβλήματα ταξινόμησης (classification) είναι $\sqrt{p}$, ενώ για προβλήματα παλινδρόμησης (regression) είναι $\frac{p}{3}$, όπου p το πλήθος των μεταβλητών του προβλήματος μας. Ωστόσο αυτός ο αριθμός δεν είναι πάντα ικανοποιητικός για το πρόβλημα μας και θα πρέπει να εξεταστεί εκ νέου αν ο αριθμός αυτός των μεταβλητών που χρησιμοποιήσαμε μας δίνει το μικρότερο OOB error στο μοντέλο που κατασκευάσαμε.

Από τα Random Forests λαμβάνουμε σημαντικά στοιχεία για το πρόβλημα μας. Ένα από τα κύρια είναι το πόσο σημαντική είναι μια μεταβλητή για την πρόβλεψη της μεταβλητής απόκρισης (Variable Importance). Για αυτό η μέθοδος αυτή μπορεί να χαρακτηριστεί όπως και η μέθοδος Lasso, ως μια μέθοδος επιλογής μεταβλητών. Επίσης, μας δίνει την διαφορά που έχει στο MSE του μοντέλου μας η αφαίρεση μιας μεταβλητής. Επιπλέον, ένα σημαντικό προτέρημα, που έχουν τα Random Forests σε σχέση με άλλες μεθόδους, είναι ότι αντιμετωπίζει καλύτερα προβλήματα στα οποία λείπουν ορισμένα δεδομένα.

Τέλος η μέθοδος των Random Forests έχει ένα σημαντικό μειονέκτημα, ότι δεν είναι εύκολα ερμηνεύσιμη. Ενώ τα δέντρα παλινδρόμησης είναι πολύ εύκολο να ερμηνευθούν σε έναν τρίτο με κόστος όμως την ακρίβεια πρόβλεψης τους, τα Random Forests είναι αρκετά ακριβή στην πρόβλεψη, αλλά πιο δύσκολο να ερμηνευθούν σε κάποιον τρίτο.

3.7 Boosting

Μια ακόμη μέθοδος που παρουσιάζει βελτίωση στην μέθοδο των δέντρων παλινδρόμησης είναι η μέθοδος Boosting, που όπως και η μέθοδος Bagging βρίσκει εφαρμογές και σε άλλους τομείς πέρα από τα δέντρα παλινδρόμησης. Σε αντίθεση με την μέθοδο Bagging που παράγει πολλά δέντρα παλινδρόμησης, τα οποία έχουν εξελιχθεί πλήρως, η μέθοδος boosting προσπαθεί να κάνει ένα αδύναμο μοντέλο σε ένα δυνατό μοντέλο. Παίρνει, δηλαδή, το ένα δέντρο παλινδρόμησης που είναι ένα μοντέλο αδύναμο, όπως εξηγήσαμε, ως προς την προβλεπτική του ικανότητα και προσπαθεί να την ενισχύσει. Αρχικά, έχουμε ένα μοντέλο έτοιμο και προσαρμόζουμε ένα δέντρο αποφάσεων στους συντελεστές αυτού του μοντέλου αντί να χρησιμοποιήσουμε την μεταβλητή απόκρισης. Στην συνέχεια προσθέτουμε αυτό το δέντρο παλινδρόμησης στο αρχικό μας μοντέλο για να πάρουμε νέους συντελεστές. Αυτήν την διαδικασία την επαναλαμβάνουμε για πολλά δέντρα αποφάσεων, τα οποία ορίζουμε εξ αρχής πόσα θα είναι. Ορίζουμε επίσης και το πόσο μεγάλα είναι σε βάθος και συνήθως επιλέγουμε δέντρα βάθους 1 τα οποία ονομάζονται stump trees. Τέλος, όταν προσθέτουμε το δέντρο στο μοντέλο μας, το προσθέτουμε επιβάλλοντας του μια ποινή λ, την οποία και αυτήν αποφασίζουμε από πριν και είναι συνήθως μικρή, συνήθεις τιμές είναι το 0.01 και το 0.001, αλλά αλλάζει από μοντέλο σε μοντέλο ανάλογα την περίπτωση (James et al., 2021). Με αυτόν τον τρόπο, αντί να παράγουμε πολλά δέντρα όπως στις μεθόδους Bagging και Random Forest, ουσιαστικά ενισχύουμε την απόδοση ενός δέντρου προσθέτοντας συνέχεια νέα δέντρα.

3.8 Διασταυρωμένη Επικύρωση (Cross Validation)

Τέλος, θα μιλήσουμε για την μέθοδο Διασταυρωμένης Επικύρωσης (Cross-Validation), η οποία είναι αυτή που μας βοηθάει να επιλέξουμε την τιμή της ρυθμιστικής παραμέτρου στις μεθόδους Ridge, Lasso και Pruning. Επίσης η μέθοδος cross validation είναι η καταλληλότερη μέθοδος για να εξετάσουμε το σφάλμα γενίκευσης ενός μοντέλου. Μπορεί δηλαδή να χρησιμοποιηθεί για να υπολογίσουμε πως συμπεριφέρεται το μοντέλο που κατασκευάσαμε σε άλλα σύνολα δεδομένων που δεν έχουν τις ίδιες μεταβλητές.

Οι δύο πιο συνήθεις μέθοδοι Cross Validation είναι η K-fold Cross Validation και η Leave-One-Out cross validation που θα αναλυθούν στης Παραγράφους 3.8.1 και 3.8.2 αντίστοιχα.

3.8.1 K-fold Cross Validation

Η K-fold cross validation λειτουργεί ως εξής: Αρχικά διαχωρίζει το σύνολο δεδομένων σε k ξεχωριστές πτυχές (folds), έτσι ώστε τα τμήματα αυτά να είναι περίπου ίσα μεταξύ τους. Στην συνέχεια, επιλέγεται ένα από αυτά τα τμήματα, το οποίο το αφήνουμε εκτός τους συνόλου δεδομένων. Παίρνουμε τα υπόλοιπα k-1 τμήματα, προσαρμόζουμε ένα μοντέλο με αυτά και χρησιμοποιώντας το αποτέλεσμα αυτού του μοντέλου, εξετάζουμε το πόσο ακριβές είναι στην πρόβλεψη του, πάνω στο τμήμα δεδομένων, που αποκόψαμε στην αρχή. Ουσιαστικά, υπολογίζουμε το MSE των παρατηρήσεων του τμήματος, που αφήσαμε εκτός στην αρχή, χρησιμοποιώντας το αποτέλεσμα του μοντέλου που προσαρμόσαμε στα υπόλοιπα k-1 τμήματα. Επαναλαμβάνουμε αυτήν την διαδικασία για κάθε ένα από τα k τμήματα (ή πτυχές) που χωρίσαμε το σύνολο δεδομένων μας στην αρχή και υπολογίζουμε τον μέσο όρο του συνολικού MSE από αυτά τα μοντέλα, δηλαδή η εκτιμήτρια της μεθόδου δίνεται από την σχέση (James et al.,2021)

$$CV_{(k)} = \frac{1}{k} \sum_{i=1}^{k} MSE_i \qquad (3.10)$$

Με αυτόν τον τρόπο μπορούμε να επιλέξουμε και την ρυθμιστική παράμετρο λ για τις μεθόδους Ridge, Lasso και Pruning. Ουσιαστικά επιλέγουμε την τιμή λ η οποία οδηγεί το μοντέλο μας στο μικρότερο πιθανό σφάλμα πρόβλεψης.

Αξίζει να σημειώσουμε ότι οι πιο συνήθεις επιλογές για την τιμή k είναι k=5 ή k=10. Γενικά δεν υπάρχει κάποιος κανόνας, ωστόσο αυτές οι δύο τιμές, εμπειρικά οδηγούν σε αποτελέσματα, που δεν πάσχουν ούτε από υψηλή μεροληψία ούτε υψηλή διασπορά και παράλληλα είναι υπολογιστικά απλές. (James et al.,2021)

3.8.2 Leave-One-Out Cross Validation

Η Leave-One-Out cross validation είναι μια υποπερίπτωση της K-fold cross validation με k=n όπου n οι παρατηρήσεις που έχουμε στο σύνολο δεδομένων μας. Ο τρόπος με τον οποίο λειτουργεί είναι ο εξής: Αρχικά χωρίζουμε το σύνολο δεδομένων μας σε n τμήματα και προσαρμόζουμε το μοντέλο μας χρησιμοποιώντας τα n-1 από αυτά. Αφού προσαρμόσουμε το μοντέλο μας, εξετάζουμε το σφάλμα του μοντέλου αυτού πάνω στην παρατήρηση που δεν χρησιμοποιήσαμε, δηλαδή το MSE του. Επαναλαμβάνουμε αυτήν την μέθοδο για όλες τις παρατηρήσεις και χρησιμοποιούμε το μέσο των σφαλμάτων που υπολογίσαμε ως την εκτιμήτρια αυτής της μέθοδού για το γενικευμένο σφάλμα του μοντέλου μας και δίνεται από την σχέση (James et al.,2021)

$$CV_{(k)} = \frac{1}{n} \sum_{i=1}^{n} MSE_i \qquad (3.11)$$

Γενικά η μέθοδος αυτή είναι σχετικά «ακριβή» σε υπολογιστικό κόστος όταν έχουμε πολλές παρατηρήσεις στο σύνολο δεδομένων μας. Ωστόσο θεωρείται ως σχεδόν αμερόληπτη εκτιμήτρια του σφάλματος (Cawley & Talbot, 2010). Τείνουμε να χρησιμοποιούμε την μέθοδο Leave-One-Out έναντι της K-fold μόνο όταν έχουμε μικρό αριθμό παρατηρήσεων.

4 Εφαρμογή

4.1 Παρουσίαση προβλήματος

Στο ποδόσφαιρο τα στατιστικά τα οποία έχουν όλοι οι φίλαθλοι στο νου τους όταν θέλουν να αξιολογήσουν έναν ποδοσφαιριστή είναι τα γκολ, οι ασσιστς και ο χρόνος συμμετοχής του. Παράλληλα τα τελευταία χρόνια στο ποδόσφαιρο έχουν παρουσιαστεί κάποια πιο προηγμένα στατιστικά που έχουν ως στόχο να ποσοτικοποιήσουν αυτά που βλέπει ένας έμπειρος scouter και ένας έμπειρος τεχνικός διευθυντής όταν παρακολουθούν έναν παίχτη. Ο στόχος αυτής της εργασίας είναι η κατασκευή ενός στατιστικού μοντέλου που εκτιμά τον αναμενόμενο μισθό ενός ποδοσφαιριστή της πρώτης κατηγορίας του Ιταλικού πρωταθλήματος λαμβάνοντας υπόψιν μόνο αγωνιστικά κριτήρια και παράλληλα να βρούμε την συσχέτιση του μισθού ενός ποδοσφαιριστή με τα στατιστικά που προαναφέραμε.

Πριν ξεκινήσουμε την στατιστική ανάλυση στα δεδομένα μας θα αναφέρουμε συνοπτικά τι είναι οι όροι xG, xA, npxG, xGChain, xGBuildup και Key Passes.

xG ή Expected Goals είναι ένα στατιστικό το οποίο έχει ως σκοπό να αξιολογεί κάθε σουτ που παίρνει ένας παίχτης και να εκτιμεί την πιθανότητα που έχει αυτό να γίνει γκολ. Οι τιμές που παίρνει το xG είναι απο 0 εώς 1 όπου το 0 σημαίνει οτί το σουτ δεν έχει καμία πιθανότητα να καταλήξει γκολ, και 1 σημαίνει ότι το σουτ αυτό είναι θεωρητικά γκολ.

xA ή Expected Assists είναι ένα στατιστικό το οποίο έχει ως στόχο να αξιολογεί κάθε πάσα που ο αποδέκτης της σουτάρει και να εκτιμάει την πιθανότητα που έχει αυτή η πάσα να γίνει ασσιστ. Η τιμές που παίρνει το xA είναι απο 0 εώς 1 όπου το 0 σημαίνει ότι η πάσα δεν έχει καμία πιθανότητα να γίνει ασσιστ, και 1 σημαίνει ότι η πάσα αυτή είναι θεωρητικά ασσιστ.

Key Passes είναι οι πάσες οι οποίες καταλήγουν σε σουτ από τον δέκτη της πάσας αλλά όχι σε γκολ.

npxG είναι ένα στατιστικό το οποίο έχει ως σκοπό να μετράει τα xG χωρίς να λαμβάνει υπόψιν τα πέναλτι. Ένα πέναλτι έχει xG=0.79 καθώς είναι γνωστό ότι το 79% των πέναλτι καταλήγουν σε γκολ. Οπότε το στατιστικό αυτό μετράει το xG ενός παίχτη μέσα στην σεζόν, στην δική μας περίπτωση χωρίς να λαμβάνει υπόψιν τα πέναλτι που εκτέλεσε.

xGChain είναι ένα στατιστικό το οποίο λαμβάνει υπόψιν όλους τους παίχτες που συμμετείχαν σε μια αλυσίδα πασών πριν από ένα σουτ χωρίς κάποια παρέμβαση αντιπάλου και καταλογίζει το xG του σουτ σε όλους αυτούς τους παίκτες. Αυτός ο αριθμός είναι το xGChain.

xGBuildup είναι ένα στατιστικό το οποίο λαμβάνει υπόψιν όλους τους παίχτες που συμμετείχαν σε μια αλυσίδα πασών πριν από ένα σουτ χωρίς κάποια παρέμβαση αντιπάλου και καταλογίζει το xG του σουτ σε όλους αυτούς τους παίκτες χωρίς ωστόσο να το προσμετρά σε αυτούς που έκαναν το τελικό σουτ ή σε αυτούς που έβγαλαν ασσιστ. Ο αριθμός αυτός είναι το xGBuildup

Για τα δεδομένα μας λάβαμε υπόψιν παίχτες που δεν είναι τερματοφύλακες και συμμετείχαν πάνω από 360 αγωνιστικά λεπτά την σεζον 2021-2022. Στο σύνολο 294 παίκτες του Ιταλικού πρωταθλήματος

Τα δεδομένα τα οποία χρησιμοποιήθηκαν προήλθαν από τις εξής πηγές

- Τα στατιστικά των παικτών πέρα από τον μισθό και την ηλικία είναι από το understat.com μέσω της βιβλιοθήκης της R worldfootballR
- Η ηλικία και ο μισθός των παικτών από το capology.com

Οι μεταβλητές του προβλήματος μας φαίνονται αναλυτικά στον πίνακα 4.1

Πίνακας 4.1: Οι μεταβλητές του προβλήματος μας

goals: Ο συνολικός αριθμός goal που είχε ένας παίχτης μέσα στην σεζόν 2021-2022 ανά 90 λεπτά **xG**: Ο αναμενόμενος αριθμός goal που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **assists**: Ο αριθμός ασσιστ που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **xA**: Ο αναμενόμενος αριθμός ασσιστ που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **shots**: Τα συνολικά σουτ που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **key_passes:** Ο αριθμός τον πασών κλειδιά που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά. **yellow_cards**: Ο αριθμός των κίτρινων καρτών που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **red_cards:** Ο αριθμός των κόκκινων καρτών που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **npg**: Ο αριθμός των γκολ χωρίς πέναλτυ που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **npxG:** Ο αναμενόμενος αριθμός των γκολ χωρίς πέναλτυ που είχε ένας παίχτης την σεζόν 2021-2022 ανά 90 λεπτά **xGChain:** Το xGChain των παιχτών την σεζόν 2021-2022 ανά 90 λεπτά **xGBuildup**: Το xGBuildup των παιχτών την σεζόν 2021-2022 ανά 90 λεπτά **Position:** Η θέση που αγωνίζεται ο παίχτης (Αμυντικός, Κεντρικός, Επιθετικός) **age:** Η ηλικία του παίχτη. **WeeklySalary:** Ο εβδομαδιαίος μισθός των παιχτών σε 1000αδες ευρώ την σεζόν 2021-2022

4.2 Στατιστική Ανάλυση με πολλαπλή γραμμική παλινδρόμηση

Αρχικά προσαρμόζουμε το γραμμικό μας μοντέλο με μεταβλητή απόκρισης το WeeklySalary και τις μεταβλητές που προαναφέρθηκαν ως τα δεδομένα μας. Για να επιλέξουμε το καταλληλότερο μοντέλο θα προσαρμόσουμε ένα μοντέλο για κάθε πιθανό συνδυασμό των συμμεταβλητών του προβλήματος και θα επιλέξουμε αυτό με το μικρότερο AIC. Τα αποτελέσματα αυτής της μεθόδου φαίνονται στον Πίνακα 4.2

Πίνακας 4.2 Επιλογή μεταβλητών με το κριτήριο AIC και άλλα στατιστικά που μας βοηθούν στην επιλογή του καταλληλότερου μοντέλου.

predictors	rsquare	adjr	predrsq	cp	aic	sbc	msep
xG red_cards xGChain xGBuildup age	0.312663	0.30073	0.28083	2.549168	3015.168	3040.953	476595.1
xG red_cards xGBuildup age	0.306096	0.296492	0.280955	3.267824	3015.963	3038.065	479478
xG red_cards npxG xGChain xGBuildup age	0.315075	0.300756	0.277461	3.550435	3016.134	3045.602	476582.9
xG xA red_cards xGChain xGBuildup age	0.314644	0.300316	0.278468	3.728864	3016.319	3045.787	476882.8
xG key_passes red_cards xGChain xGBuildup age	0.31431	0.299975	0.277466	3.867391	3016.462	3045.931	477115.6
xG xA red_cards npxG xGChain xGBuildup age	0.318689	0.302013	0.277103	4.054631	3016.579	3049.731	475732.2
xG assists red_cards xGChain xGBuildup age	0.313818	0.299473	0.277329	4.070884	3016.673	3046.142	477457.6
xG yellow_cards red_cards xGChain xGBuildup age	0.313743	0.299397	0.277914	4.101858	3016.705	3046.174	477509.7
xG xGChain xGBuildup age	0.304202	0.294572	0.275023	4.05194	3016.765	3038.866	480786.7
xG key_passes red_cards npxG xGChain xGBuildup age	0.318168	0.30148	0.275542	4.270248	3016.803	3049.956	476095.9
xG shots red_cards xGChain xGBuildup age	0.312963	0.298599	0.270782	4.425127	3017.039	3046.508	478053.1

Στον Πίνακα 4.2 παρουσιάζονται τα 11 μοντέλα με τις χαμηλότερες τιμές AIC από τα 16384 πιθανά μοντέλα που μπορούν να κατασκευαστούν με τις μεταβλητές του προβλήματος. Στον πίνακα αυτό το rsquare συμβολίζει τον συντελεστή προσδιορισμού R^2, το adjr συμβολίζει τον διορθωμένο συντελεστή προσδιορισμού R^2_{adj}, το predrsq συμβολίζει το R^2 predicted, το cp που συμβολίζει το στατιστικό Cp-Mallows, το aic που συμβολίζει το κριτήριο AIC, το sbc που συμβολίζει το κριτήριο BIC και το msep που συμβολίζει το μέσο τετραγωνικό σφάλμα της πρόβλεψης. Το R^2 predicted μας δείχνει την προβλεπτική ικανότητα του μοντέλου μας και λειτουργεί με όμοιο τρόπο με την Leave One Out cross validation που αναλύσαμε στην Παράγραφο 3.8.2.

Αξίζει να σημειωθεί ότι για το μοντέλο μας επιλέξαμε μεταβλητές με την χρήση του AIC. Λόγω του ότι έχουμε μεγάλο αριθμό παρατηρήσεων (294) το AIC και το AIC_c δίνουν πανομοιότυπα αποτελέσματα, οπότε δεν υπάρχει λόγος να χρησιμοποιήσουμε το AIC_c.

Από τα αποτελέσματα λοιπόν του Πίνακα 4.2 καταλήγουμε ότι το καλύτερο μοντέλο είναι αυτό που περιγράφεται από τις μεταβλητές xG, red_cards, xGChain, xGBuildup, age. Καθώς από τον Πίνακα 4.2 παρατηρούμε ότι πέρα από την μικρότερη τιμή AIC έχει και την μικρότερη τιμή Cp-Mallows από τα υπόλοιπα μοντέλα. Παράλληλα έχει το δεύτερο μικρότερο BIC και μέσο τετραγωνικό σφάλμα πρόβλεψης και τέλος ένα ικανοποιητικό R^2. Και για αυτό το μοντέλο έχουμε τα αποτελέσματα του πίνακα 4.3. Από τα οποία παρατηρούμε ότι το xGChain και το red_cards δεν είναι στατιστικά σημαντικές καθώς το p-value τους είναι μεγαλύτερο του 0.05, ωστόσο αξίζει να σημειωθεί ότι έχουν αρκετά χαμηλό p-value κοντά στο 0.05. Επίσης το ποσοστό της διασποράς που εξηγείται από το μοντέλο μας είναι το 31.27% σύμφωνα με τον συντελεστή προσδιορισμού R^2. Ωστόσο αξίζει να σημειώσουμε ότι σύμφωνα με τον R^2 predicted το ποσοστό αυτό είναι 28.08%.

Το ίδιο μοντέλο παίρνουμε ως κατάλληλο και με την χρήση της μεθόδου της διαδοχικής αφαίρεσης χρησιμοποιώντας ως κριτήριο το F-Test. Τα δύο τελευταία βήματα που λαμβάνουμε είναι τα εξής:

```
Step:  AIC=3016.13
WeeklySalary ~ xG + red_cards + npxG + xGChain + xGBuildup +
    age

              Df  Deviance    AIC       F value     Pr(>F)
- npxG        1   466846     3015.2     1.0109     0.31554
<none>            465208     3016.1
- red_cards   1   470418     3017.4     3.2144     0.07405 .
- xGChain     1   471138     3017.9     3.6583     0.05678 .
- xG          1   472294     3018.6     4.3718     0.03742 *
- xGBuildup   1   492582     3030.9    16.8879     5.179e-05 ***
- age         1   507270     3039.6    25.9495     6.361e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Step:  AIC=3015.17
WeeklySalary ~ xG + red_cards + xGChain + xGBuildup + age

             Df     Deviance    AIC       F value    Pr(>F)
<none>               466846     3015.2
- xGChain     1     471307     3016.0     2.7516     0.098244 .
- red_cards   1     472593     3016.8     3.5453     0.060724 .
- xG          1     484100     3023.8    10.6436     0.001237 **
- xGBuildup   1     502732     3034.9    22.1382     3.944e-06 ***
- age         1     511392     3040.0    27.4806     3.075e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Call:  glm(formula = WeeklySalary ~ xG + red_cards + xGChain +
xGBuildup +
    age, data = SerieAFootballers)

Coefficients:
(Intercept)           xG    red_cards      xGChain    xGBuildup
age
    -70.146       70.382     -168.069       29.059        3.923
3.008

Degrees of Freedom: 293 Total (i.e. Null);  288 Residual
Null Deviance:      679200
Residual Deviance: 466800  AIC: 3015
```

Πίνακας 4.3: Μοντέλο πολλαπλής γραμμικής παλινδρόμησης.

Συντελεστές	estimate	std.error	t.statistic	p.value
(Intercept)	-70.146	15.4932	-4.528	<0.001
xG	70.382	21.5733	3.262	0.00124
red_cards	-168.069	89.2614	-1.883	0.06072
xGChain	29.059	17.5181	1.659	0.09824
xGBuildup	3.923	0.8338	4.705	<0.001
age	3.008	0.5739	5.242	<0.001

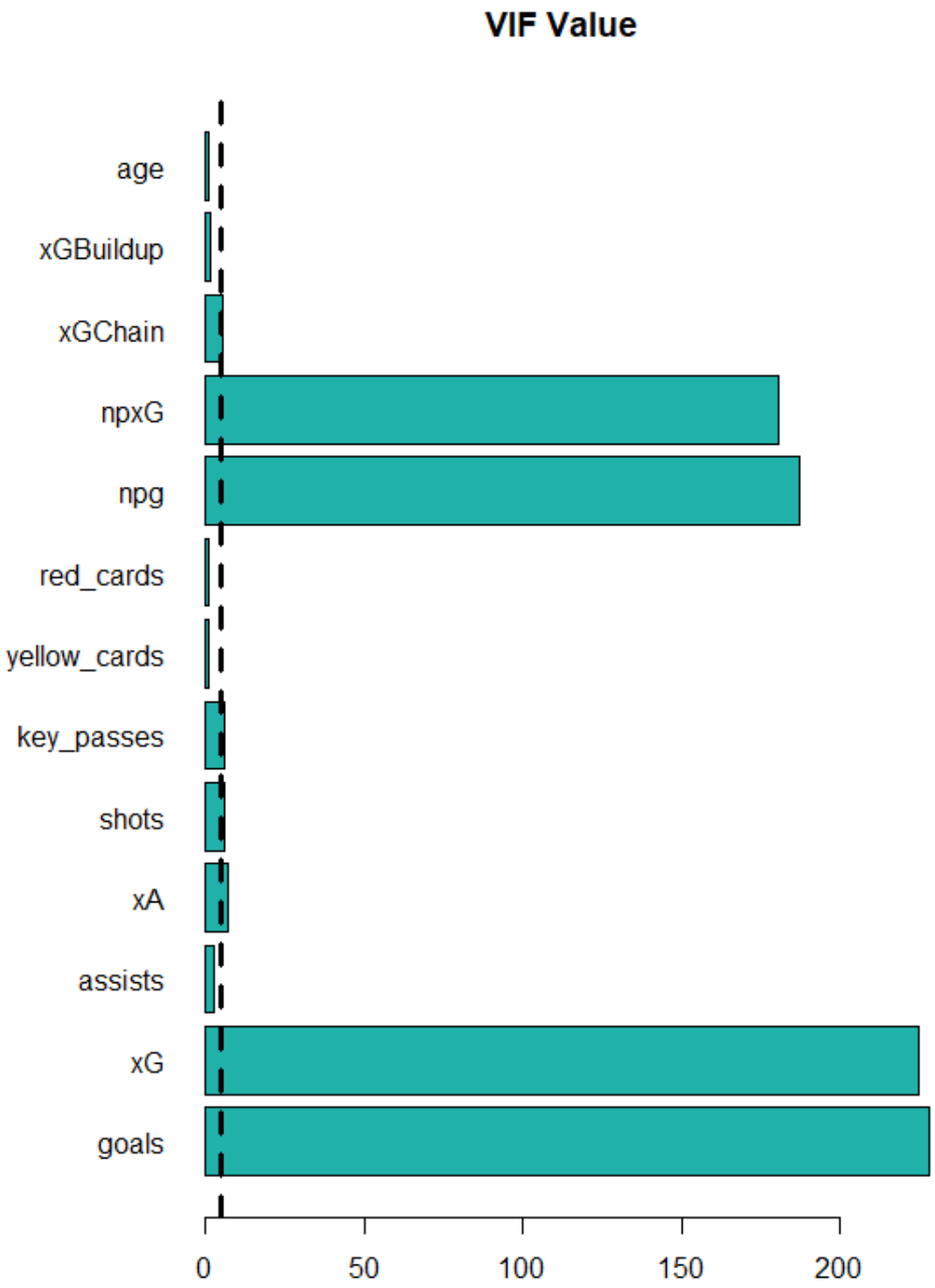
Multiple R-squared	0.3127

Στην συνέχεια θα εξετάσουμε την τιμή VIF των μεταβλητών του μοντέλου η οποία μας δείχνει το κατά πόσο θα αυξηθεί η διακύμανση ενός εκτιμώμενου συντελεστή εάν η αντίστοιχη ανεξάρτητη μεταβλητή παρουσιάζει πολυσυγγραμικότητα. Αυτό το κάνουμε για να εξετάσουμε την πολυσυγγραμικότητα του μοντέλου μας καθώς γνωρίζουμε ότι όταν έχουμε πολυσυγκραμικότητα δεν πρέπει να χρησιμοποιούμε την μέθοδο AIC καθώς η επιλογή του μοντέλου μας δεν θα είναι ιδανική. Ως κανόνα έχουμε ότι αν η τιμή VIF είναι μικρότερη του 1 δεν υπάρχει συγγραμικότητα, αν η τιμή είναι ανάμεσα από το 1 και το 5 υπάρχει σχετική συγγραμικότητα και αν η τιμή είναι μεγαλύτερη από το 5 υπάρχει συγγραμικότητα.

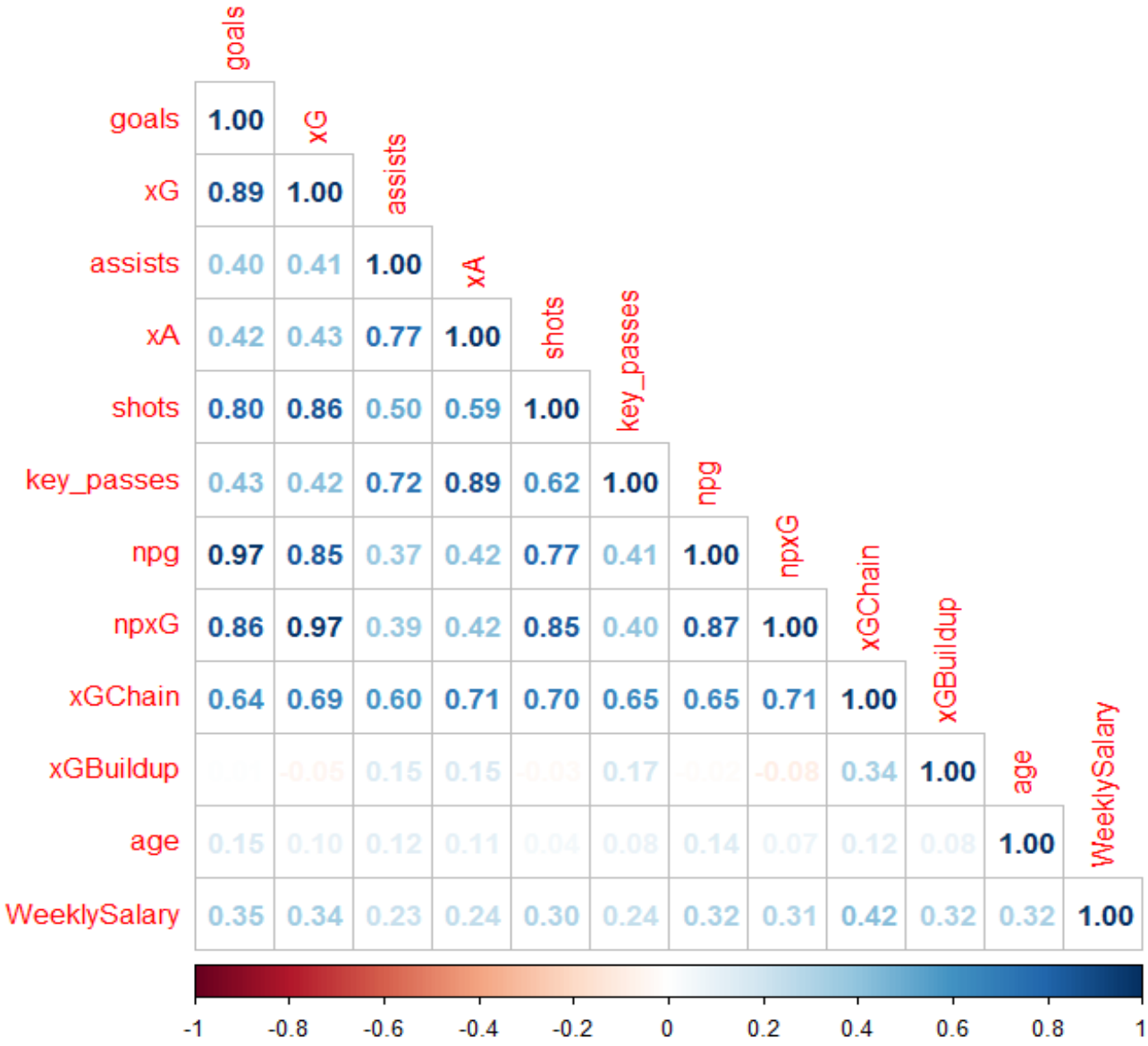
Πίνακας 4.4: VIF values του μοντέλου μας

Μεταβλητές	**VIF**
goals	228
xG	225.0773
assists	2.592819
xA	7.094247
shots	6.157567
key_passes	5.825507
yellow_cards	1.128589
red_cards	1.047033
npg	187.4412
npxG	180.4127
xGChain	5.532198
xGBuildup	1.756296
age	1.077923

Όπως φαίνεται από τις τιμές VIF των μεταβλητών μας υπάρχει το πρόβλημα της πολυσυγγραμικότητας στο μοντέλο μας καθώς έχουμε μεταβλητές που ξεπερνάνε το 5. Στο Γράφημα 4.2 θα εξετάσουμε και την συγγραμικότητα που έχουν οι μεταβλητές μεταξύ τους ανά δύο. Όσο πιο κοντά είναι η τιμή στο 1 τόσο πιο συγγραμικές.



Γράφημα 4.1: Γραφική αναπαράσταση των VIF value με διαχωριστική γραμμή στο 5



Γράφημα 4.2 Συσχέτιση των συμμεταβλητών ανα 2.

Εύκολα βλέπουμε από το Γράφημα 4.2 ότι οι περισσότερες μεταβλητές του μοντέλου είναι συγγραμικές μεταξύ τους ανά δύο.

Πλέον μπορούμε να συμπεράνουμε ότι η προσαρμογή του γενικού γραμμικού μοντέλου μας και η επιλογή μεταβλητών με το κριτήριο AIC δεν είναι οι ιδανικές μέθοδοι λόγω της πολυσυγγραμικότητας του μοντέλου μας.
Για αυτόν τον λόγο λοιπόν θα χρησιμοποιήσουμε τις μέθοδούς Ridge και Lasso για την προσαρμογή του γενικού γραμμικού μοντέλου των δεδομένων μας και τις μεθόδους Lasso και Δέντρα Παλινδρόμησης, Τυχαία δάση,bagging και boosting, για την επιλογή των μεταβλητών μας και την κατασκευή του τελικού μας μοντέλου.

4.3 Παλινδρόμηση Κορυφογραμμής (Ridge Regression)

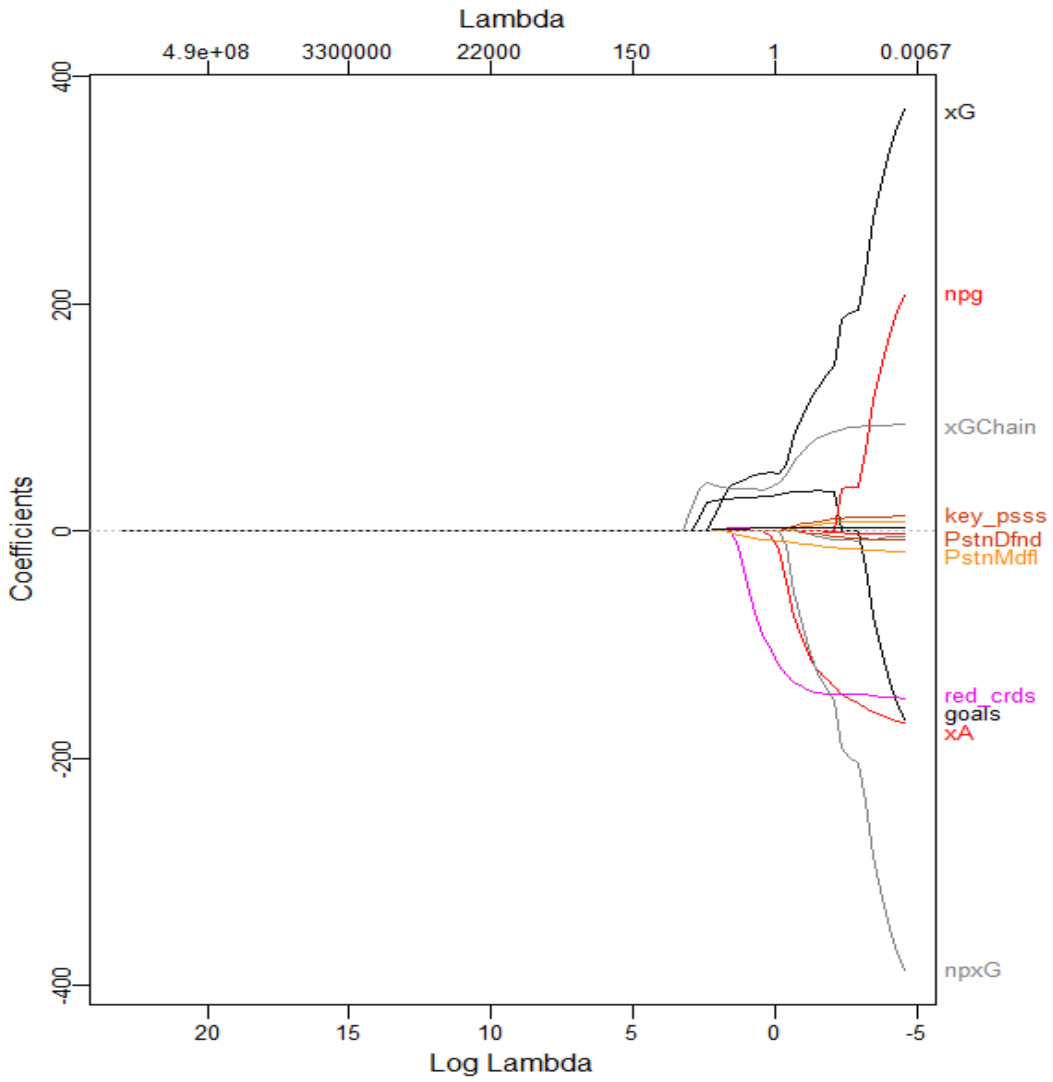
Αφού κατασκευάσουμε το σύνολο εκπαίδευσης μας και προσαρμόσουμε την παλινδρόμηση Ridge σε αυτό παίρνουμε το MSE στην περίπτωση που θέλουμε να προσαρμόσουμε ένα μοντέλο πολλαπλής γραμμικής παλινδρόμησης σαν αυτό που παρουσιάσαμε πριν (θα κάνουμε μια Ridge παλινδρόμηση με λ=0) και το αποτέλεσμα είναι **MSE= 1601.385.** Τέλος υπολογίζουμε την καλύτερη τιμή λ για το μοντέλο μας, η οποία είναι **λ= 34.15378** και υπολογίζουμε το MSE του test που έχουμε με την παλινδρόμηση Ridge το οποίο είναι **MSE= 1526.801.** Οπότε μπορούμε να συμπεράνουμε ότι το μοντέλο που προσαρμόσαμε με την μέθοδο Ridge είναι καταλληλότερο από τις προαναφερθείς μεθόδους.
Το μοντέλο που παίρνουμε για λ= 34.15378 είναι αυτό του πίνακα 4.5

Πίνακας 4.5 Συντελεστές παλινδρόμησης Ridge

(Intercept)	**-28.1631961**
goals	**15.3459332**
xG	**19.3710833**
assists	**5.8443205**
xA	**3.2991596**
shots	**1.3506941**
key_passes	**0.3481671**
yellow_cards	**-11.2731400**
red_cards	**-90.6000502**
npg	**10.1732354**
npxG	**11.9750564**
xGChain	**24.2041746**
xGBuildup	**2.3151834**
PositionDefender	**2.2543575**
PositionMidfielder	**-2.0906955**
age	**1.7953884**

4.4 Παλινδρόμηση Lasso

Όπως παρατηρούμε (το γνωρίζουμε και από την θεωρία) η μέθοδος Ridge δεν είναι μέθοδος επιλογής μεταβλητών καθώς ποτέ δεν μηδενίζει κάποια από αυτές. Οπότε θα εφαρμόσουμε τώρα την παλινδρόμηση Lasso από την οποία περιμένουμε να πάρουμε μια παλινδρόμηση όπου κάποιοι συντελεστές θα είναι 0. Αρχικά σχεδιάζουμε την γραφική παράσταση των συντελεστών της παλινδρόμησης Lasso και παίρνουμε το αποτέλεσμα του Γραφήματος 4.3.



Γράφημα 4.3 Γραφική παράσταση συντελεστών της Lasso για το ποιοί μηδενίζονται

Παρατηρούμε ότι αρκετοί συντελεστές είναι κοντά στο 0.

Στην συνέχεια υπολογίζουμε το βέλτιστο λ για την παλινδρόμηση Lasso με όμοιο τρόπο με την Ridge (δημιουργία training set) και καταλήγουμε ότι είναι το **λ= 2.069573** και υπολογίζουμε το MSE το οποίο είναι **MSE= 1491.119** και παρατηρούμε ότι είναι κοντά σε αυτό που υπολογίσαμε με την παλινδρόμηση

Ridge αλλά αποτελεί μια βελτίωση. Τέλος θα πάρουμε την ακόλουθη παλινδρόμηση Lasso στον πίνακα 4.6.

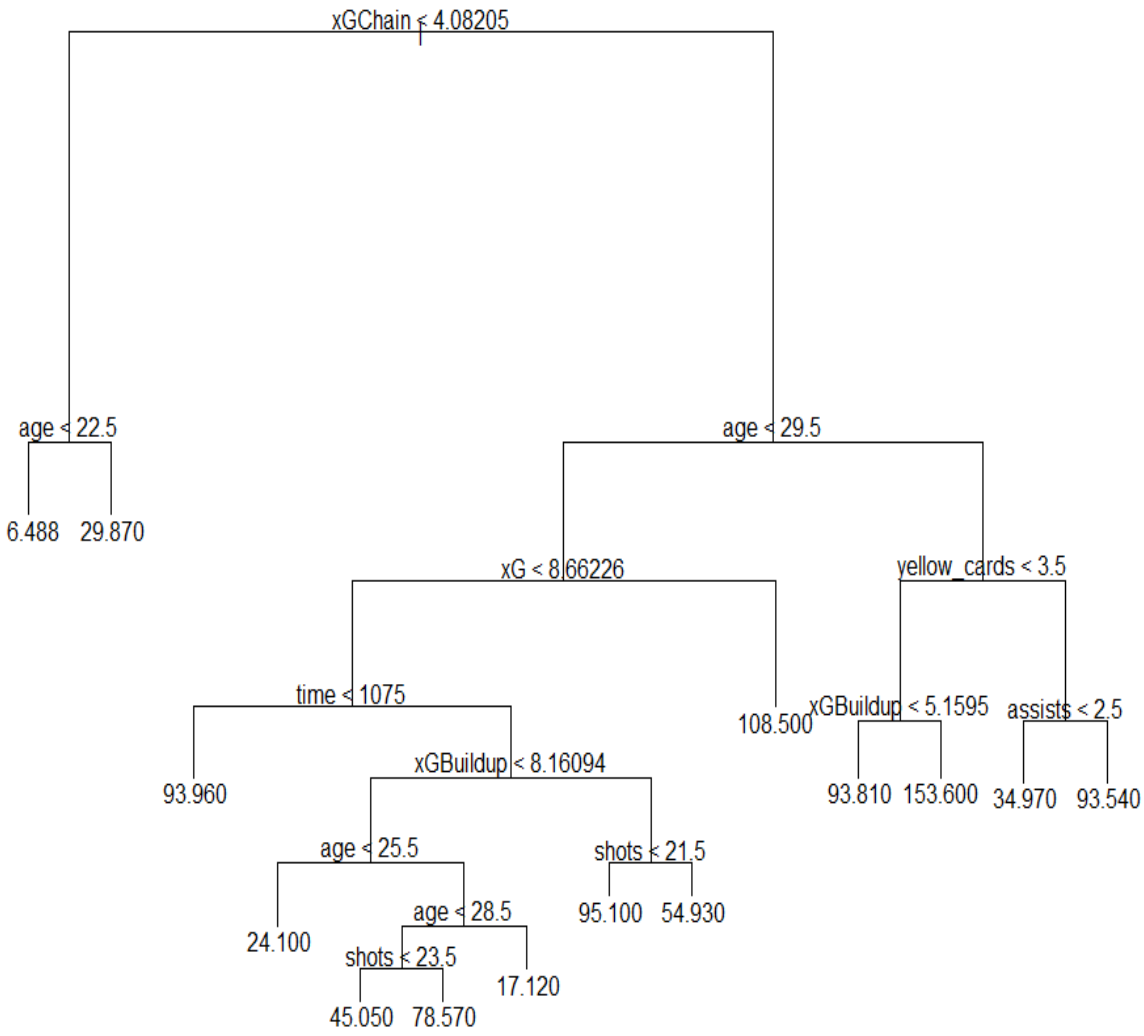
Πίνακας 4.6 Συντελεστές παλινδρόμησης Lasso

(Intercept)	-53.75454564
goals	7.620818
xG	47.339907
assists	0
xA	0
shots	0
key_passes	0
yellow_cards	-1.984451
red_cards	-87.563042
npg	0
npxG	0
xGChain	32.199666
xGBuildup	3.226166
PositionDefender	0
PositionMidfielder	-1.558087
age	2.557920

Καταλήγουμε ότι τα οι assists, οι xA, τα shots,οι key_passes,τα npg,τα npxG και το PositionDefender ενός ποδοσφαιριστή δεν παίζουν κάποιο ρόλο στον εβδομαδιαίο μισθό του ενός ποδοσφαιριστή της Ιταλίας, ωστόσο τα στατιστικά όπως τα goals, xG, το xGChain και το xGBuildup, το yellow_cards,το red_cards και το age παίζουν ρόλο. Τέλος το ποσοστό της διασποράς που εξηγείται από το μοντέλο μας είναι το 30. 76307% μια μικρή μείωση σε σχέση με αυτό της απλής γραμμικής παλινδρόμησης.

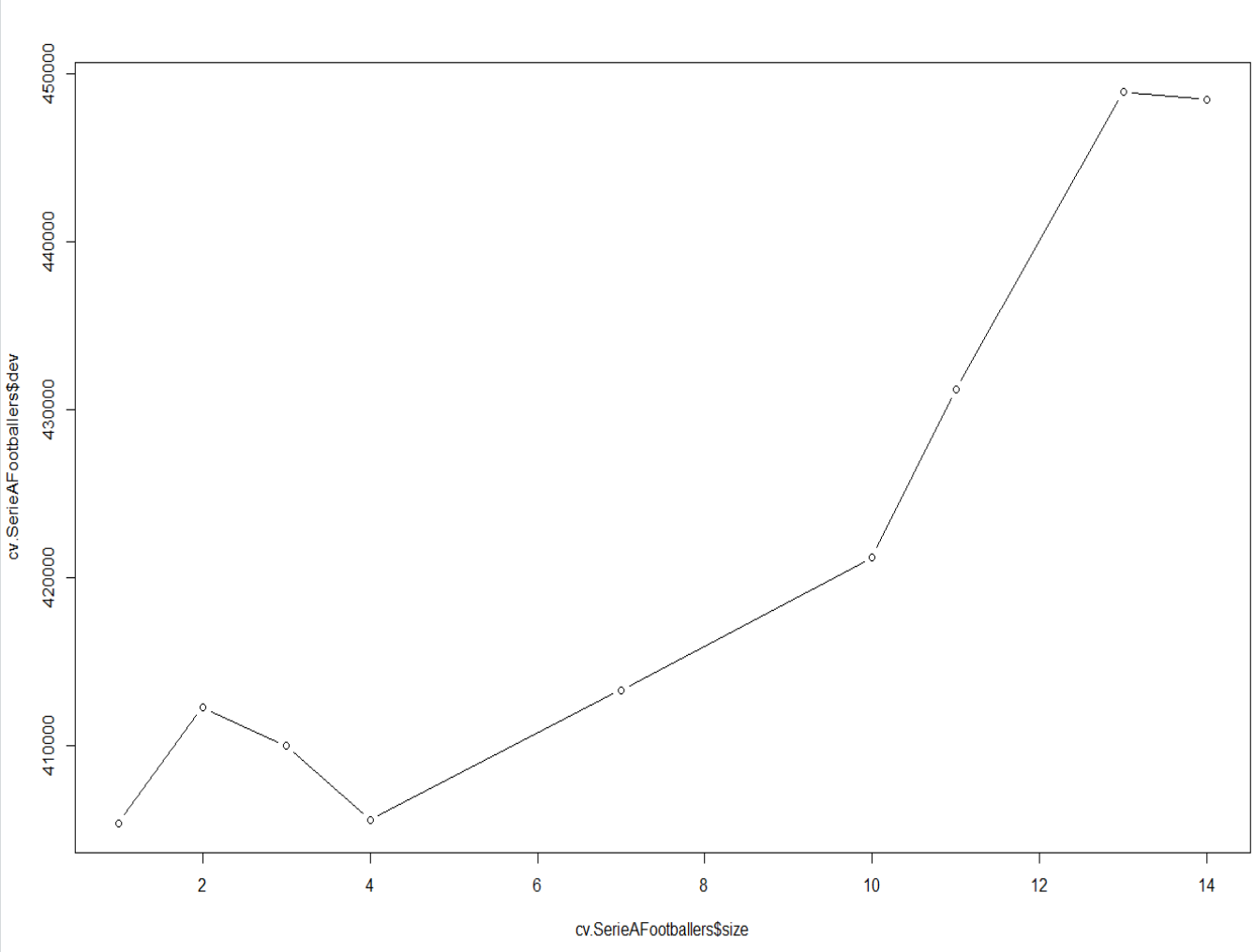
4.5 Decision Trees, Random Forest, Bagging και Boosting

Για την γραμμική παλινδρόμηση με μεταβλητή απόκρισης το WeeklySalary και την κατασκευή του δένδρου παλινδρόμησης της έχουμε ότι έχουν χρησιμοποιηθεί 8 από τις συμμεταβλητές του προβλήματος μας και λαμβάνουμε το δέντρο παλινδρόμησης του πίνακα Γραφήματος 4.4



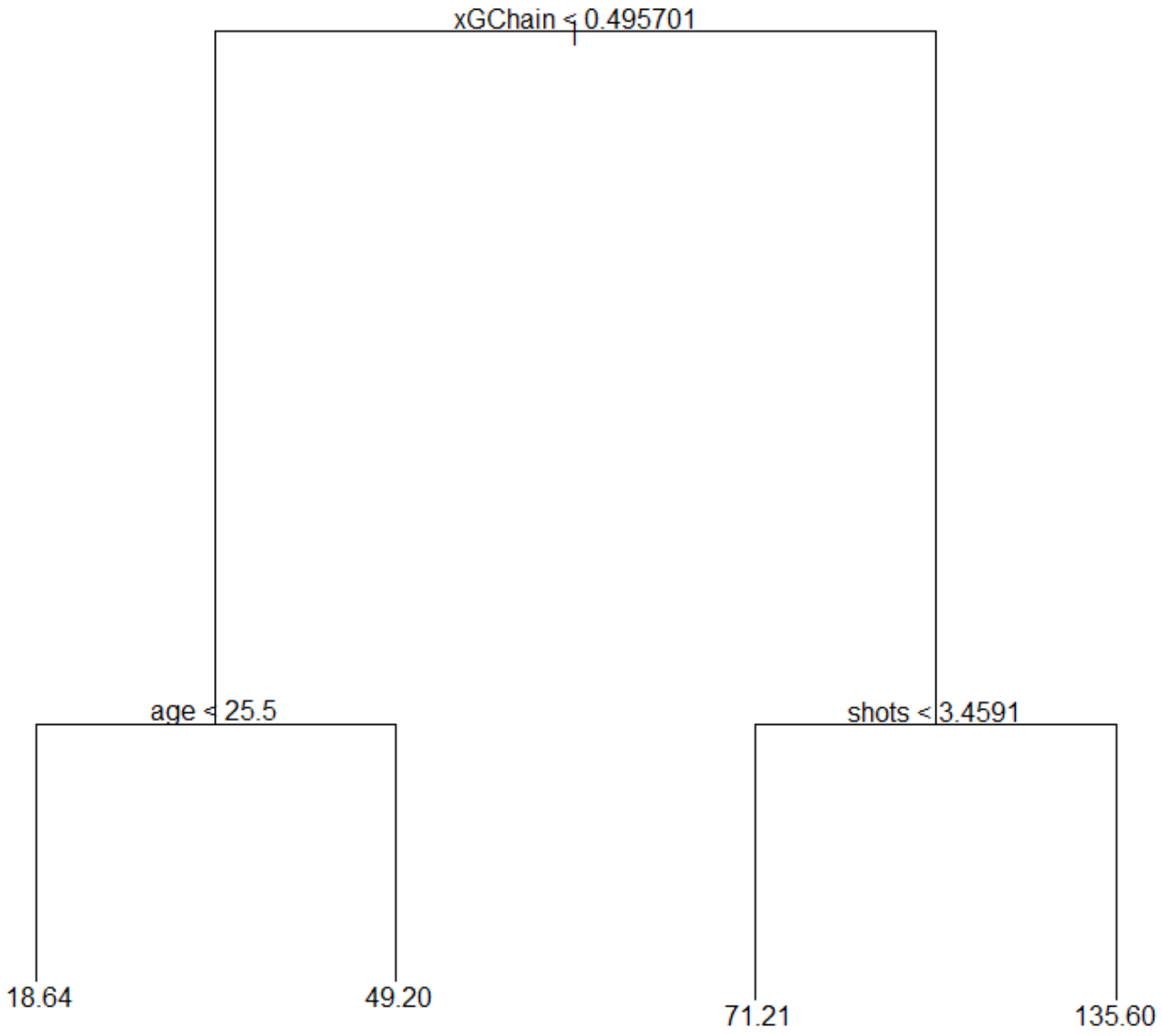
Γράφημα 4.4 Δένδρο παλινδρόμησης χωρίς κάποια βελτίωση

Το δέντρο το οποίο λαμβάνουμε μας δίνει 14 τελικά νούμερα το οποίο το κάνει δύσκολο στην κατανόηση. Για αυτόν τον λόγο θα ελέγξουμε αν μπορούμε να καλυτερεύσουμε αυτό το δέντρο και παράλληλα να μικρύνουμε τα τελικά αποτελέσματα που λαμβάνουμε με την μέθοδο pruning και χρησιμοποιώντας την μέθοδο cross validation όπως εξηγήσαμε στην Παράγραφο 3.4. Σχεδιάζουμε μια γραφική παράσταση που παρουσιάζει το cross validation error συναρτήσει του μεγέθους του δέντρου. Και όπως καταλαβαίνουμε από την παρακάτω γραφική παράσταση του Γραφήματος 4.5 το ιδανικό μέγεθος του δέντρου είναι 4.



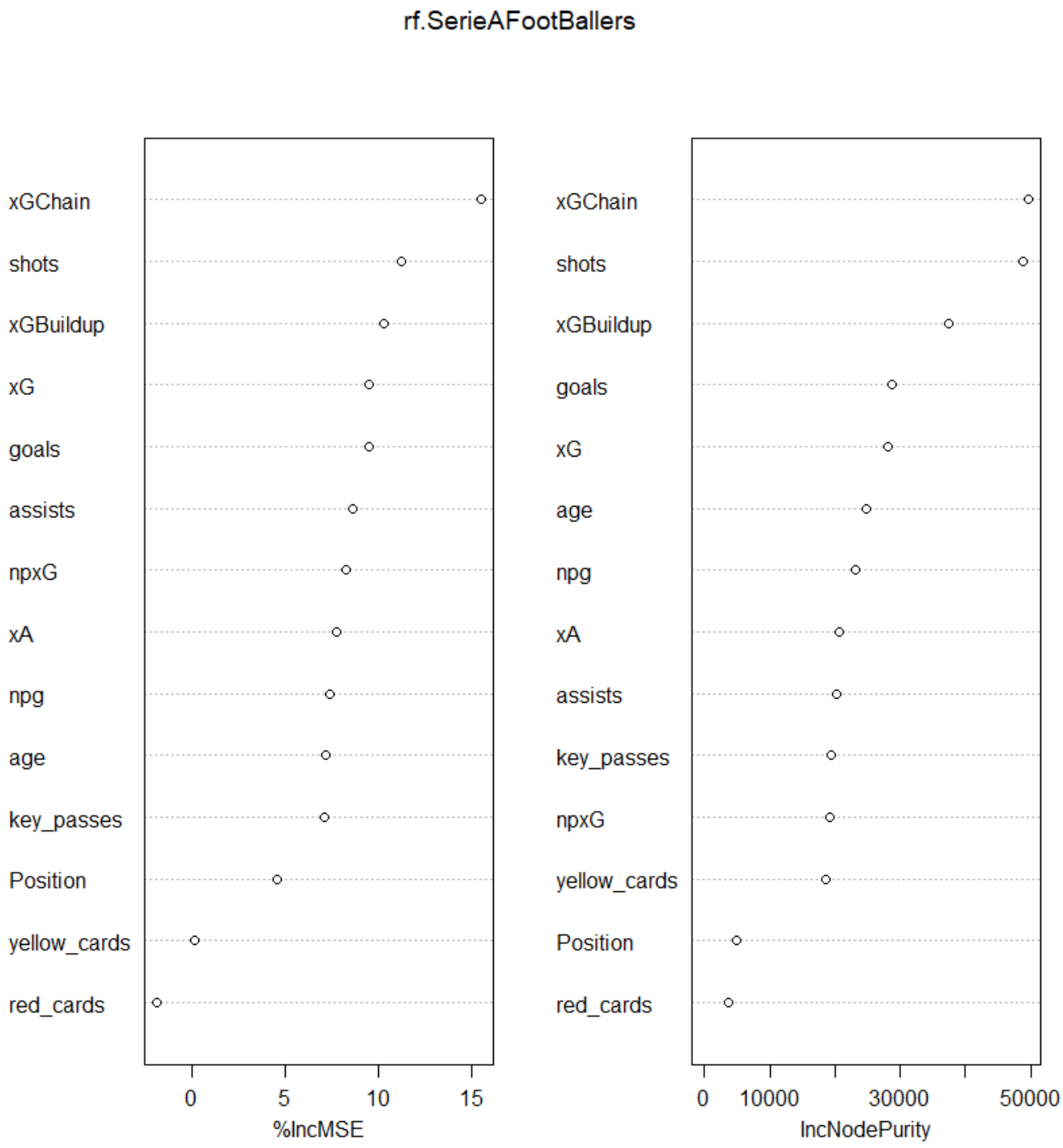
Γράφημα 4.5 Cross Validation Error συναρτήσει μεγέθους του δέντρου παλινδρόμησης

Και το δέντρο που λαμβάνουμε για την τιμή 4 είναι το ακόλουθο στο Γράφημα 4.6



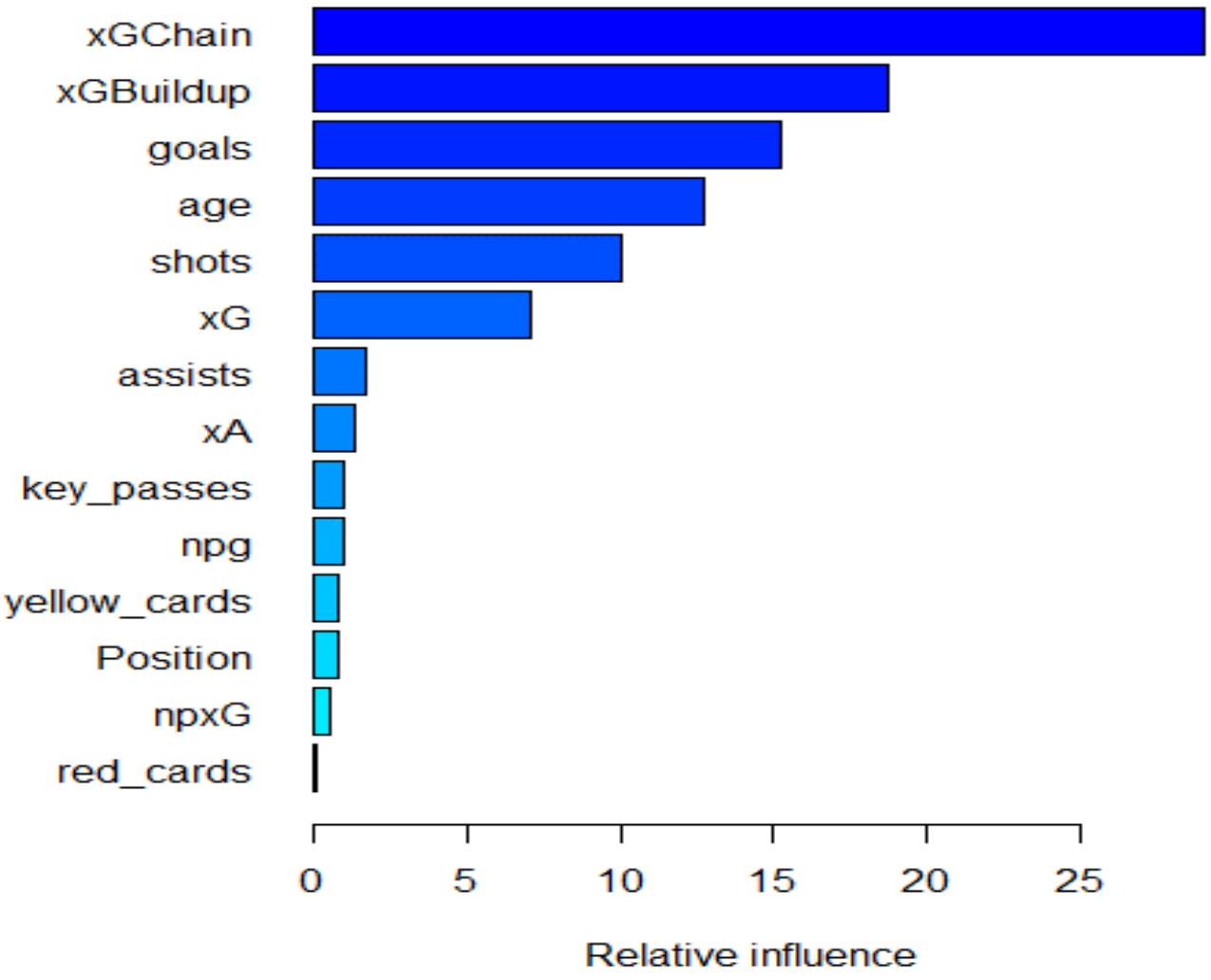
Γράφημα 4.6 Δέντρο παλινδρόμησης μεγέθους 4

Όπως καταλαβαίνουμε προφανώς αυτό το αποτέλεσμα δεν είναι επιθυμητό καθώς είναι ανακριβές οπότε θα χρησιμοποιήσουμε την μέθοδο των τυχαίων δασών για να εντοπίσουμε ποιες μεταβλητές είναι σημαντικές στο μοντέλο μας. Κατασκευάζουμε το τυχαίο δάσος μας θέτοντας ως αριθμό δέντρων στο τυχαίο δάσος τα 1000 δέντρα και ως αριθμό μεταβλητών του κάθε δέντρου το 5 το οποίο, ο αριθμός των μεταβλητών επιλέχθηκε καθώς αυτός ο αριθμός μας δίνει το μικρότερο Out of Bag error που αποτελεί το σημαντικότερο κριτήριο καταλληλότητας των μεθόδων Random Forests, Bagging και Boosting (Breiman,2001) και αφού εφαρμόσουμε και την μέθοδο bagging με τις ίδιες συνθήκες παρατηρούμε ότι το τυχαίο δάσος μας δίνει καλύτερο μέσο τετραγωνικό σφάλμα 1766.52 έναντι 2261.867 της μεθόδου Bagging. Όπως ήταν αναμενόμενο δεν παρατηρούμε βελτίωση του τυχαίου δάσους. Στον παρακάτω πίνακα μπορούμε να δούμε ποιες μεταβλητές παρατηρούμε ότι είναι οι σημαντικότερες στο τυχαίο δάσος που κατασκευάσαμε. Αξίζει να σημειωθεί ότι όσο πιο δεξιά στον άξονα χ είναι η άσπρη μπάλα τόσο πιο σημαντική είναι η μεταβλητή. Το αριστερά διάγραμμα μας δείχνει την επιρροή της μεταβλητής στο μέσο τετραγωνικό σφάλμα του μοντέλου μας άμα την αφαιρέσουμε, ενώ το δεξιά διάγραμμα μας δείχνει την μέση μείωση του Gini index, μεγάλη τιμή της μέσης μείωσης του Gini index (IncNodePurity) μας δείχνει ότι η μεταβλητή είναι πιο σημαντική από κάποια με μικρότερη τιμή. Επίσης το μοντέλο των τυχαίων δασών μπορεί να εξηγήσει κατά 15.74% την διασπορά του δείγματος μας, ενώ το μοντέλο μας με την μέθοδο bagging κατά 15.06%



Γράφημα 4.7 Σημαντικότητα των μεταβλητών στο μοντέλο των Random Forests

Παρατηρούμε από το Γράφημα 4.7 ότι η πιο σημαντική μεταβλητή είναι το xGChain. Και παρατηρούμε γενικά ότι σύμφωνα με το gini index το xGChain το shots και το xGBuildup είναι οι πιο σημαντικές μεταβλητές, ενώ αυτές που έχουν την μεγαλύτερη επιρροή στο MSE είναι το xGChain και σε λίγο μικρότερο βαθμό το shots το xGBuildup και οι υπόλοιπες. Ωστόσο λόγω του τετραγωνικού σφάλματος που έχουμε από το μοντέλο προτιμούμε να κρατήσουμε την παλινδρόμηση Lasso ως το μοντέλο μας. Στην συνέχεια θα εξετάσουμε αν μπορούμε να βελτιώσουμε ακόμα περισσότερο το μοντέλο των τυχαίων δασών με την χρήση της μεθόδου boosting η οποία μας δίνει μέσο τετραγωνικό σφάλμα 2051.669 που είναι μια βελτίωση από το μοντέλο με την μέθοδο Bagging ωστόσο όχι από το μοντέλο με την χρήση της μεθόδου Random Forest.



Γράφημα 4.8 Σημαντικότητα μεταβλητών με την μέθοδο boosting.

Στο Γράφημα 4.8 βλέπουμε την επιρροή των μεταβλητών σύμφωνα με την μέθοδο Boosting και παρατηρούμε ότι οι σημαντικότερες μεταβλητές στο μοντέλο μας είναι το xGChain και το xGBuildup. Ωστόσο δεν θα επιλέξουμε αυτό το μοντέλο γιατί το μέσο τετραγωνικό σφάλμα του είναι μεγαλύτερο από της Lasso και του Random Forest.

4.6 Τελικό Μοντέλο, Παρατηρήσεις και Συμπεράσματα

Συνοψίζοντας, λόγω του φαινομένου της πολυσυγγραμικότητας που εμφανίζεται στα δεδομένα μας και της ακρίβειας πρόβλεψης από το MSE αλλά και του ποσοστού της διασποράς που επεξηγούν οι επεξηγηματικές μεταβλητές μας το μοντέλο που θα επιλέξουμε είναι αυτό που μας δίνει η παλινδρόμηση Lasso πιο συγκεκριμένα αυτό του πίνακα 4.6. Το μοντέλο μας μας δείχνει ότι οι κύριοι παράγοντες του μοντέλου μας είναι τα goals, το xG, τα yellow_cards, τα red_cards, το xGChain, το xGBuildup, το age και το PositionMidfielder και ουσιαστικά αυτοί είναι από τα στατιστικά που είχαμε οι παράγοντες που συνεισφέρουν στον μισθό ενός ποδοσφαιριστή. Το μοντέλο μας μπορεί να εξηγήσει την διασπορά του μισθού των ποδοσφαιριστών κατά 30.76307% ωστόσο αυτό ήταν κάτι αναμενόμενο καθώς ασχοληθήκαμε μόνο με το αγωνιστικό κομμάτι και καθόλου με το πόσο εμπορικός είναι ένας παίκτης ή από ποιόν ατζέντη εκπροσωπείται. Επίσης ο συντελεστής διασποράς του μισθού των ποδοσφαιριστών στο σύνολο δεδομένων μας είναι 1.029832>1 οπότε το σύνολο δεδομένων μας έχει υψηλή διασπορά. Λαμβάνοντας τα παραπάνω υπόψιν μπορούμε να καταλήξουμε ότι ενώ το μοντέλο μας δεν είναι ιδανικό από μόνο του για να προβλέψει τον μισθό ενός ποδοσφαιριστή της 1^{ης} κατηγορίας του Ιταλικού πρωταθλήματος, είναι ένα ικανοποιητικό μοντέλο για να προβλέψει τον μισθό του όταν μας ενδιαφέρει μόνο η αγωνιστική του απόδοση.

Μια χρήσιμη εφαρμογή αυτού του μοντέλου είναι η εύρεση παιχτών που αυτήν την στιγμή αμείβονται πολύ χαμηλότερα από τον αναμενόμενο μισθό τους βάσει των αγωνιστικών κριτηρίων και προσέγγισης τους για μεταγραφή με μια δελεαστικότερη προσφορά ένα παράδειγμα τέτοιου ποδοσφαιριστή είναι ο Gianluca Scamacca της Sasuollo ο οποίος σύμφωνα με το μοντέλο μας θα έπρεπε να αμείβεται με 66,687 χιλιάδες ευρώ την εβδομάδα ωστόσο πληρώνεται μόνο με 3,654 χιλιάδες ευρώ την εβδομάδα.

Άλλη μια περίπτωση που μπορεί να χρησιμοποιηθεί το μοντέλο αυτό είναι όταν ένας παίκτης ζητάει ένα βελτιωμένο συμβόλαιο μπορούμε να δούμε αν οι απαιτήσεις του είναι λογικές και συμβαδίζουν με το μοντέλο μας όσον αφορά την αγωνιστική του επίδοση. Ένα παράδειγμα είναι ο Rafael Leao της Milan όπου αμείβεται με 34.423 χιλιάδες ευρώ την εβδομάδα και ζητάει να αμείβεται με 137 χιλιάδες ευρώ την εβδομάδα, το μοντέλο μας τον κοστολογεί στα 53,4 χιλιάδες ευρώ την εβδομάδα οπότε έχοντας στο νου μας την αγωνιστική του επίδοση οι απαιτήσεις του είναι εξωπραγματικές και θα πρέπει η ομάδα να λάβει υπόψιν και τα έσοδα που φέρνει στην ομάδα αυτός ο παίκτης ώστε να πάρει την τελική της απόφαση.

5 Βιβλιογραφία

Α) Διεθνής Βιβλιογραφία

Akaike, H. (1974) A New Look at the Statistical Model Identification. *IEEE Transactions on Automatic Control*, AC-**19**, 716-723. doi:10.1109/TAC.1974.1100705

Breiman, L., Friedman, J., Stone, C.,, Olshen, R. (1984). *Classification and Regression Trees*. Chapman and Hall/CRC.

Breiman, L. (1996). Bagging predictors. *Machine Learning* **24**, 123–140. doi:10.1007/bf00058655

Breiman, L. (2001). Random Forests. *Machine Learning* **45**, 5–32. doi:10.1023/A:1010933404324

Burnham, K. P. & Anderson, D. R. (2002). *Model Selection and Inference: A Practical Information-Theoretic Approach* (2nd ed.). Springer-Verlag New York Inc..

Cawley, G. & Talbot, N. (2010). On Over-fitting in Model Selection and Subsequent Selection Bias in Performance Evaluation. *Journal of Machine Learning Research*. **11**, 2079-2107.

Chin, W.W. (1998). The Partial Least Squares Approach to Structural Equation Modeling. *Modern Methods for Business Research*, **2**, 295-336.

Hair, J. F., Black, W. C., Babin, B. J., & Anderson, R. E. (2010). *Multivariate Data Analysis.* Prentice Hall.

Hair, J. F., Ringle, C., Sarstedt, M. (2011). PLS-SEM: Indeed a silver bullet. *The Journal of Marketing Theory and Practice*. **19**. 139-151. doi:10.2753/MTP1069-6679190202.

Hair, J. F., Ringle, C. Sarstedt, M. (2013). Partial Least Squares Structural Equation Modeling: Rigorous Applications, Better Results and Higher Acceptance. *Long Range Planning*. **46**. 1-12. doi:10.1016/j.lrp.2013.08.016.

Hanusz, Z., Tarasinska, J., Zieliński, W. (2016). Shapiro–Wilk Test with Known Mean. *Revstat Statistical Journal*. **14**, 89-100. doi:10.57805/revstat.v14i1.180.

Hastie, T., Tibshirani, R., Wainwright, M. (2016). *Statistical Learning with Sparsity-The Lasso and Generalizations*. Chapman and Hall/CRC.

James, G., Witten, D., Hastie, T., Tibshirani, R. (2021). *An Introduction to Statistical Learning: with Applications in R* (2nd ed.). Springer.

Kock, N. & Lynn, G. (2012). Lateral Collinearity and Misleading Results in Variance-Based SEM: An Illustration and Recommendations. *Journal of the Association of Information Systems*. **13**, 546-580. doi:10.17705/1jais.00302

Lawrence,T. (2018,August 30). *Introducing xGChain and xGBuildup*. Statsbomb. https://statsbomb.com/articles/soccer/introducing-xgchain-and-xgbuildup/

Louppe, G. (2014). *Understanding Random Forests: From Theory to Practice.* doi:10.13140/2.1.1570.5928. Ph.D. thesis, University of Liege

Miller, A.J. (2002). *Subset Selection in Regression* (2nd ed.), Champan and Hall New York.

Silver, N. (2012). *The signal and the noise: The art and science of prediction.* London: Allen Lane.

Zivkovic, J. (2022). worldfootballR. https://CRAN.R-project.org/package=worldfootballR

Β) Ελληνική Βιβλιογραφία

Καρώνη, Χ. & Οικονόμου, Π. (2017). *Στατιστικά Μοντέλα Παλινδρόμησης: Με χρήση MINITAB και R* (2η εκδ.). Αθήνα: Εκδόσεις ΣΥΜΕΩΝ

Φουσκάκης, Δ. (2013) *Ανάλυση Δεδομένων με Χρήση της R*, Αθήνα: Εκδόσεις ΤΣΟΤΡΑΣ.

6 Παραρτήματα

6.1 Παράρτημα Ι (Κώδικας που Χρησιμοποιήθηκε)

Εισαγωγή βιβλιοθηκών που θα χρειαστούμε

```
library(worldfootballR)
library(dplyr)
library(glmnet)
library(plotmo)
library(readxl)
library(gbm)
library(olsrr)
library(writexl)
library(caTools)
library(car)
library(quantmod)
library(corrplot)
library(tree)
library(randomForest)
options(scipen=999)
```

Εισαγωγή των δεδομένων μας στην R με την βιβλιοθήκη worldfootballR και τα στοιχεία από το capology που είναι σε μορφή excel

```
SerieA1 <- understat_team_players_stats(team_url =
c("https://understat.com/team/AC_Milan/2021","https://understat.com/team/Napoli/2021","
https://understat.com/team/Inter/2021","https://understat.com/team/Juventus/2021","https:
//understat.com/team/Atalanta/2021","https://understat.com/team/Roma/2021","https://un
derstat.com/team/Lazio/2021","https://understat.com/team/Fiorentina/2021","https://unders
tat.com/team/Verona/2021","https://understat.com/team/Sassuolo/2021","https://understat.
com/team/Torino/2021",
"https://understat.com/team/Bologna/2021","https://understat.com/team/Empoli/2021","htt
ps://understat.com/team/Udinese/2021","https://understat.com/team/Spezia/2021","https://
understat.com/team/Sampdoria/2021","https://understat.com/team/Cagliari/2021","https://u
nderstat.com/team/Venezia/2021","https://understat.com/team/Genoa/2021","https://under
stat.com/team/Salernitana/2021")
dplyr::glimpse(SerieA1)
SerieA2 <- read_excel("SerieASalary.xlsx")
```

Τροποποίηση των δεδομένων μας ώστε να έχουμε ένα ενιαίο σύνολο δεδομένων καθώς τα δεδομένα προήλθαν από δύο διαφορετικές πηγές οπότε ενδέχεται να έχουμε είτε παίκτες που να λείπει ο μισθός τους στο σύνολο δεδομένων που έχουμε ονομάσει SerieA1 είτε παίκτες που λείπουν τα στατιστικά τους στο σύνολο δεδομένων που έχουμε ονομάσει SerieA2. Παράλληλα αφαιρούμε στήλες που δεν θα χρειαστούμε από τα δεδομένα μας. Επίσης για ευκολία αντί να έχουμε 12 διαφορετικές κατηγορίες στην μεταβλητή Position τροποποιούμε τα δεδομένα μας ώστε να έχουμε μόνο 3, τις κατηγορίες Defender, Midfielder και Attacker.

```
SerieA2<-SerieA2[,-1]
colnames(SerieA2) <- c("player_name","WeeklySalary", "AnnualSalary","Position","Age")
SerieA2<-
data.frame(SerieA2$player_name,SerieA2$Position,SerieA2$Age,SerieA2$WeeklySalary)
colnames(SerieA2) <- c("player_name","Position","age", "WeeklySalary")
SerieAFootballers<-inner_join(SerieA1,SerieA2, by  ='player_name')
SerieAFootballers<-distinct(SerieAFootballers,player_name, .keep_all= TRUE)
SerieAFootballers <- select(SerieAFootballers,-player_id,-position,-season,-team_name)
rownames(SerieAFootballers)<-SerieAFootballers[,1]
SerieAFootballers<-select(SerieAFootballers,-player_name)
SerieAFootballers$Position <- gsub('CF|RW|SS|LW', 'Attacker', SerieAFootballers$Position)
SerieAFootballers$Position <-
```

```
gsub('CM|AM|DM|RM|LM','Midfielder',SerieAFootballers$Position)
SerieAFootballers$Position <- gsub('CB|LB|RB','Defender',SerieAFootballers$Position)
```

Αφαιρούμε από το σύνολο δεδομένων μας τους τερματοφύλακες και όσους έχουν χρόνο συμμετοχής στην σεζόν μικρότερο από 360 λεπτά και ανάγουμε τα στατιστικά σε στατιστικά ανά 90 λεπτά και τον μισθό σε ευρώ ανά 1000αδες

```
SerieAFootballers<-SerieAFootballers %>%
 mutate(
  across(c(3:13),
        .fns = ~./time))
SerieAFootballers<-SerieAFootballers %>%
 mutate(
  across(c(3:13),
        .fns = ~.*90))
SerieAFootballers<-filter(SerieAFootballers, time > 360)
SerieAFootballers<-filter(SerieAFootballers, Position!="GK")
SerieAFootballers <- SerieAFootballers %>% mutate(WeeklySalary = WeeklySalary/1000)
SerieAFootballers$Position <- as.factor(SerieAFootballers$Position)
SerieAFootballers<-SerieAFootballers[,-1]
SerieAFootballers<-SerieAFootballers[,-1]
```

Προσαρμόζουμε με την μέθοδο των ελαχίστων τετραγώνων το μοντέλο με όλες τις μεταβλητές αλλά και χωρίς την μεταβλητή Position.

```
Model_All_Variables<-glm(WeeklySalary~.,data=SerieAFootballers) #GLM
summary(Model_All_Variables)
Model_All_Variables2<-lm(WeeklySalary~.,data=SerieAFootballers)
ModelWithout_Position<-glm(WeeklySalary~.-Position,data=SerieAFootballers)
summary(ModelWithout_Position)
```

Εφαρμόζουμε την μέθοδο της διαδοχικής αφαίρεσης με κριτήριο το F-test για να επιλέξουμε τις μεταβλητές του μοντέλου

```
step(Model_All_Variables, direction = "backward", test="F")
```

Εξετάζουμε τις τιμές AIC, BIC, Cp-Mallows, Rsquared, Rsquared predicted και MSE predicted για όλους τους πιθανούς συνδυασμούς των συμμεταβλητών του μοντέλου και προσαρμόζουμε το μοντέλο με την καλύτερη τιμή AIC λαμβάνοντας ωστόσο υπόψιν και τα υπόλοιπα στατιστικά και λαμβάνουμε τα αποτελέσματα των πινάκων 4.2 και 4.3

Πίνακας 4.2

```
TestAllModels<-data.frame(ols_step_all_possible(Model_All_Variables2))
TestAllModels
```

Πίνακας 4.3

```
ModelWith_xG_redcards_xGChain_xGBuildup_age<-glm(WeeklySalary~.-assists -xA -shots -key_passes -npg -yellow_cards -Position -npxG -goals,data=SerieAFootballers)
summary(ModelWith_xG_redcards_xGChain_xGBuildup_age)
```

Υπολογίζουμε την τιμή VIF και δημιουργούμε τα Γραφήματα 4.1 και 4.2 και τον πίνακα 4.4

Πίνακα 4.4

```
vif(ModelWithout_Position)
```

Γράφημα 4.1

```
par(mar=c(4,7,4,4))

barplot(vif(ModelWithout_Position), main = "VIF Value", horiz = TRUE, col = "lightseagreen",las=1)

abline(v = 5, lwd = 3, lty = 2) #Vif Values
```

Γράφημα 4.2

```
var<-cor(select(SerieAFootballers,-Position))

corrplot(var,method='number',type='lower') #Correlation charts of variables. Numbers
```

Προσαρμόζουμε το μοντέλο μας με την μέθοδο της Παλινδρόμησης Ridge

```
x=model.matrix(WeeklySalary~.,SerieAFootballers)[,-1]
y=SerieAFootballers$WeeklySalary
grid=10^seq(10,-2,length=100)
set.seed(1)
train=sample(1:nrow(x),nrow(x)/2)
test=(-train)
y.test=y[test] #Training Set
ridge.mod=glmnet(x[train,],y[train],alpha=0,lambda = grid,thresh=1e-12)
```

Υπολογισμός του MSE της μεθόδου Ελαχίστων Τετραγώνων μέσω της μεθόδου Ridge, δηλαδή θέτοντας λ=0

```
ridge.pred=predict(ridge.mod,s=0,newx=x[test,],exact=T,x=x[train,],y=y[train])
mean((ridge.pred-y.test)^2)#MSE Least Square.
```

Χρήση της μεθόδου Διασταυρωμένης Επικύρωσης για την κατάλληλη επιλογή λ για την μέθοδο Ridge

```
set.seed(1)
cv.ridge=cv.glmnet(x[train,],y[train],alpha=0)
plot(cv.ridge)
bestlam=cv.ridge$lambda.min
bestlam #BestLambda selection
```

Προσαρμογή της Παλινδρόμησης Ridge με την τιμή λ που υπολογίσαμε και κατασκευή του Πίνακα 4.5 και υπολογισμός του MSE της.

```
ridge.pred=predict(ridge.mod,s=bestlam,newx=x[test,])
mean((ridge.pred-y.test)^2) #MSE με BestLambda
Ridge=glmnet(x,y,alpha=0)
predict(Ridge,type="coefficients",s=bestlam)[1:16,]# Ridge Regression με BestLambda
RidgeSerieA<-predict(Ridge,type="coefficients",s=bestlam)[1:16,]
```

Προσαρμόζουμε το μοντέλο μας με την μέθοδο της Παλινδρόμησης Lasso και κατασκευάζουμε το Γράφημα 4.3

```
lasso.mod=glmnet(x[train,],y[train],alpha=1,lambda=grid)
plot_glmnet(lasso.mod)#coefficient plot
```

Χρήση της μεθόδου Διασταυρωμένης Επικύρωσης για την κατάλληλη επιλογή λ για την μέθοδο Lasso

```
set.seed(1)
cv.Lasso=cv.glmnet(x[train,],y[train],alpha=1)
plot(cv.Lasso)
bestlam=cv.Lasso$lambda.min#BestLambda selection
bestlam
```

Προσαρμογή της μεθόδου Lasso με την τιμή λ που υπολογίσαμε, υπολογισμός MSE και κατασκευή του πίνακα 4.6

```
lasso.pred=predict(lasso.mod,s=bestlam,newx=x[test,])
mean((lasso.pred-y.test)^2)
Lasso=glmnet(x,y,alpha=1,lambda=bestlam)
```

```
lasso.coef=predict(Lasso,type="coefficients",s=bestlam,standarize=FALSE)[1:16,]
lasso.coef #Lasso Regression coefficients
```

Υπολογισμός του ποσοστού της διασποράς που εξηγείται από το μοντέλο μας

```
Lasso$dev.ratio
```

Κατασκευή ολόκληρου δέντρου παλινδρόμησης και δημιουργία του Γραφήματος 4.4

```
set.seed(1)
train=sample(1:nrow(SerieAFootballers),nrow(SerieAFootballers)/2)
tree.SerieAFootballers<-tree(WeeklySalary~.,data=SerieAFootballers,subset= train)
summary(tree.SerieAFootballers)
plot(tree.SerieAFootballers)
text(tree.SerieAFootballers,pretty=0)
```

Χρήση της μεθόδου της Διασταυρωμένης Επικύρωσης για την επιλογή κατάλληλης τιμής α για να εφαρμόσουμε την μέθοδο pruning και δημιουργία του Γραφήματος 4.5 που μας δείχνει τον ιδανικό αριθμό του μεγέθους του δείγματος μας και κατασκευή αυτού του δέντρου στο Γράφημα 4.6

Γράφημα 4.5

```
cv.SerieAFootballers<-cv.tree(tree.SerieAFootballers)
plot(cv.SerieAFootballers$size,cv.SerieAFootballers$dev,type="b")
```

Γράφημα 4.6

```
prune.SerieAFootballers<-prune.tree(tree.SerieAFootballers,best=4)
plot(prune.SerieAFootballers)
text(prune.SerieAFootballers)
```

Υπολογισμός MSE του δέντρου παλινδρόμησης

```
yhat<-predict(tree.SerieAFootballers,newdata=SerieAFootballers[-train,])
SerieAFootballers.test<-SerieAFootballers[-train,"WeeklySalary"]
mean((yhat-SerieAFootballers.test)^2)
```

Μέθοδος Bagging με 1000 δέντρα

```
bag.SerieAFootballers<-
randomForest(WeeklySalary~.,data=SerieAFootballers,subset=train,mtry=14,ntree=1000,impor
tance= TRUE)
bag.SerieAFootballers
yhat.bag<-predict(bag.SerieAFootballers,newdata=SerieAFootballers[-train,])
```

Μέθοδος Random Forests με 1000 δέντρα

```
rf.SerieAFootBallers<- randomForest(WeeklySalary~.,data=
SerieAFootballers,subset=train,mtry=5,ntree=1000,importance= TRUE)
rf.SerieAFootBallers
yhat.rf<-predict(rf.SerieAFootBallers,newdata=SerieAFootballers[-train,])
```

Υπολογισμός MSE της μεθόδου Random Forest

```
mean((yhat.rf-SerieAFootballers.test)^2)
```

Υπολογισμός MSE της μεθόδου Bagging

```
mean((yhat.bag-SerieAFootballers.test)^2)
```

Υπολογισμός σημαντικότητας μεταβλητών σύμφωνα με το μοντέλο που κατασκευάσαμε με την μέθοδο Bagging

```
importance(bag.SerieAFootballers)
varImpPlot(bag.SerieAFootballers)
```

Υπολογισμός σημαντικότητας μεταβλητών σύμφωνα με το μοντέλο που κατασκευάσαμε με την μέθοδο Random Forest και κατασκευή του Γραφήματος 4.7

```
importance(rf.SerieAFootBallers)
varImpPlot(rf.SerieAFootBallers)
```

Κατασκευή μοντέλου με την μέθοδο Boosting χρησιμοποιώντας 1000 δέντρα μεγέθους 1 όπως είπαμε στην θεωρία και κατασκευάζουμε το Γράφημα 4.8

```
set.seed(1)
boost.SerieAFootballers<-
gbm(WeeklySalary~.,data=SerieAFootballers[train,],distribution="gaussian",n.trees=1000,intera
ction.depth = 1,shrinkage = 0.002)
par(mar=c(5,7,12,4),las=1)
summary(boost.SerieAFootballers)
```

Υπολογισμός MSE με την μέθοδο Boosting

```
yhat.boost<- predict(boost.SerieAFootballers,newdata=SerieAFootballers[-train,],n.trees=1000)
mean((yhat.boost-SerieAFootballers.test)^2)
```