APPLYING STATISTICAL LEARNING TECHNIQUES TO ANALYZE MOTORSPORTS USING FORMULA 1 AS AN EXAMPLE

Uladzislau Luksha
uladzislau.luksha@studenti.unimi.it
ID 964000

ABSTRACT

This report applies supervised and unsupervised statistical learning techniques to a detailed analysis of salaries and race performance in the luxury motorsport sector, using Formula 1 as an example. The dataset used for this study includes important metrics for measuring the professional achievement of Formula 1 sportsmen, including, among others, the number of races held, championship titles, driver experience, and the average yearly wage over the course of a career. The dataset provided uses fictional player names, rounded sensitive metrics, and extra noise to shield personal information.

The goal of this study was to use unsupervised statistical learning to find hidden patterns in the data. PCA, K-Means, and Hierarchical Clustering techniques were employed for this. Another challenge was creating a statistical model that might be used for luxury motorsport wage prediction and inference. We employed LASSO, Linear and Random Forest Regression for this aim. The project also discusses the problem of small datasets, the dummy variable trap, and the problem of multicollinearity in regression.

1 DATASET EXPLORATORY ANALYSIS

1.1 DATASET DESCRIPTION

In this paper, we analyse a dataset of 750 players of Formula One generated in 2022 with the help of privacy preservation techniques for the sensitive data. The dataset contains 12 features, which report the number of races, years in the sport career, number of championship titles for each player, number of times each driver started the race at the first position of the track grid etc. Some of the features like ‘team’ and ‘favorite circuit’ are categorical with 10 and 24 levels simultaneously. It should be noted that the names of the players are intentionally changed to fictional for privacy purposes. The dataset also contains the average salary throughout each players carrier. This indicator is used later as a target variable during the section of supervised learning. Average annual salary is expressed in millions of USD. Given that the exact wages of each player are

unknown, this indicator was calculated with severe estimation. The dataset was uploaded from Kaggle and is advised for practicing various regression techniques [1,2]. The names of the features, their description and range are listed in the Table 1 below.

Table 1. Features of the dataset and their description.

Feature	Feature Description	Range
name	Name of the driver (privacy and masking techniques applied)	750 unique values
country	Country from which the driver comes	9 countries presented
team	The f1 team that the driver drives for the major part of their career	10 teams presented
races	Number of GPs the driver has competed in.	[20…300]
years_in_f1	Number of years a driver has spent in Formula One	[1…14]
championships	Number of Championship Titles that the driver holds	[0…8]
gp_wins	Number of races that the driver has won throughout his career.	[30…100]
podiums	Number of times a driver finished in the top-3 throughout his career.	[0…90]
poles	Number of times a driver has started in the front of the grid.	[30…100]
dnfs	Number of times the driver did not finish the race.	[5…20]
fav_circuit	A driver's favorite circuit	24 circuits presented
avg_yearly_salary	The average yearly salary that the driver earned throughout his career in millions of USD	[5...20]

1.2 DATA PREPROCESSING

After checking that our dataset does not contain the missing values, we initiate the data pre-processing. We extract all the numeric and categorical values and treat them separately. For all the numeric features we calculate the basic descriptive statistics, including mean and standard deviation. Figure 1 provides the information about the descriptive statistics of the variables.

	Mu	sigma
races	161.06	44.51
years_in_f1	7.81	2.04
championships	3.78	1.25
gp_wins	65.18	11.73
podiums	41.15	13.31
poles	66.10	11.59
dnfs	12.33	2.55
avg_yearly_salary	14.16	2.18

Figure 1. Numeric features' descriptive statistics

For the purposes of unsupervised learning (PCA, K-means and Hierarchical Clustering) we perform the standardization of our data. For the purposes of supervised learning the data is left unscaled considering the need of the regression models to be interpretable. Then we proceed with the categorical data. For the 'favorite circuit' feature we decided to create the enlarged categories considering the geographical area of the circuit location. Thus, we formed 5 groups out of 24 unique circuits: 'APAC', 'LATAM', 'NORTH_AMERICA', 'MEAST_AFR' and 'EUROPE', which stand for Asia Pacific and Australia, Latin America, North America, Middle East and Africa, and Europe simultaneously. Given there is no relationship among levels of the categorical features, for purposes of supervised part of this study we performed one-hot encoding of 'team' and 'favorite circuit' variables, having created the dummies for each level.

In the final step of data preprocessing for supervised techniques, we deal with the outliers.

1.3 OUTLIERS

As an outlier, we define a player whose performance is significantly different from the others in the dataset. And the anomalous behavior can be observed for one variable as well as for several variables. One of the most popular methods for recognizing outliers is the interquartile range. A graph of boxplots of quantitative variables with potential univariate outliers is presented in Appendix A, Figure 17. The disadvantage of the interquartile range is that it does not take into account possible outliers for multiple dimensions, which is why we will use a more complicated method called Isolation Forest.

Isolation Forest is an algorithm for finding anomalies. Instead of modeling the normal points, it uses isolation (how remote a data point is from the rest of the data) to find anomalies [13]. In R, this algorithm can be implemented with the 'solitude' library. By running the algorithm using 500 trees, we determine the threshold for an outlier equal 0.595, which contributes to 85-percentile of an

anomaly score. With this method we identify the presence of 72 outliers, which we remove from the dataset that we will use for the supervised part. Given that the anomaly score for all outliers is slightly greater than the threshold, we will not remove them from dataset for unsupervised learning purposes.

1.4 CORRELATION PLOT

After relabeling and standardizing the data, we calculated the correlation matrix among all the numeric features of the dataset. If we seek to perform the dimensionality reduction and investigate the hidden patterns of the data, there should be correlation between the variables. Figure 2 shows the correlation plot for the numeric (and newly-created dummy variables).

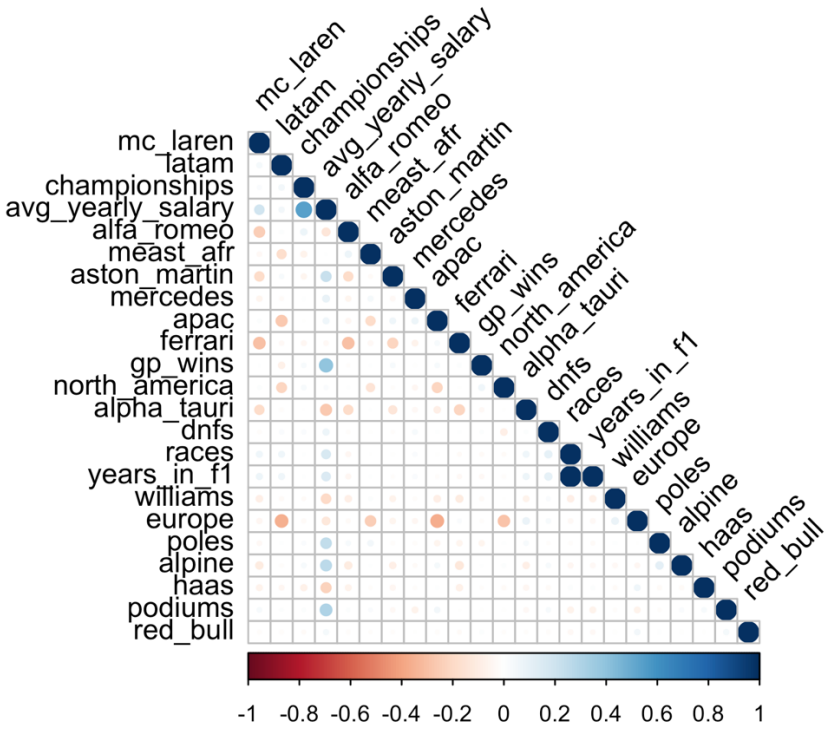

Figure 2. Correlation plot between the numeric (and dummy) variables

There is a very strong, almost functional, dependence between years spent in Formula 1 and number of races for the player. It may be simply explained that these variables are the linear combination of each other. Hence, for the purposes of regression one of them can be easily removed since it does not add any information. For the rest of the numeric variables the correlation is not so pronounced, which means that there is likely a higher number of dimensions we need to keep in order to preserve the information provided by the original data.

There is also a noticeable relationship between the dummy variables representing 'team' categorical feature, e.g., 'Alpha Romeo' and 'Ferrari', 'McLaren' and

'Ferrari' etc. Such a nuisance should be carefully treated during the feature selection for linear regression model since it creates the problem of multicollinearity.

NB: Dummy variables are not used in the unsupervised part of this study. PCA and K-Means Clustering methods are performed using only the numeric features of the model meanwhile the Hierarchical Clustering includes also the categorical features without one-hot encoding.

2 UNSUPERVISED LEARNING: PCA AND CLUSTERING

2.1 PRINCIPAL COMPONENT ANALYSIS

Principle component analysis allows to reduce the dimensionality of the data preserving the most of the existent variation. For the application of PCA we used only the original numerical data, the subset of which includes 8 out of 12 dataset features. We intentionally dropped the categorical variables since the PCA can be applied based on only numeric ones. Dimensionality reduction may be also performed on the categorical variables using MCA (multiple correspondence analysis). It allows to detect and represent underlying structures in a dataset by representing data as points in a low-dimensional Euclidean space [10]. However, that technique is out of scope of this study and may be considered for the future work.

Our principal components are the normalized eigenvectors of the covariance matrix between the features based on the data normalization carried out during the preprocessing step. The first principal component explains 26.2% of the variance in the dataset, the second component – 21.7%. Figure 3 represents eigenvalues, variance percentage explained and the cumulative variance percentage explained for 8 PCs. Since 5 principal components explain more than 85% of the variance, we can use them to perform the dimensionality reduction keeping the major part of the dataset's information.

	eigenvalue	variance.percent	cumulative.variance.percent
Dim.1	2.095770765	26.1971346	26.19713
Dim.2	1.738766137	21.7345767	47.93171
Dim.3	1.052618037	13.1577255	61.08944
Dim.4	1.020528755	12.7566094	73.84605
Dim.5	0.950023109	11.8752889	85.72134
Dim.6	0.924054478	11.5506810	97.27202
Dim.7	0.208668496	2.6083562	99.88037
Dim.8	0.009570222	0.1196278	100.00000

Figure 3. PCA eigenvalues, variance percentage explained, cumulative variance percentage explained

To be sure that the principal components are not correlated with each other, we compute the correlation plot for the PCs (Figure 4). As expected, there is no evidence of PCs mutual correlation.

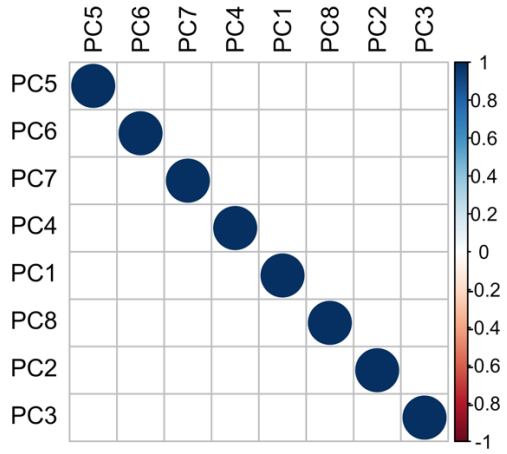

Figure 4. Correlation plot between the principal components

Next step is to study the feature contribution of to principal components. We implemented it by creation of the factor map that shows the features' contribution to the first and the second principal components (Figure 5). Features that are far from the origin are well represented by the principal components because the distance between them and the origin measures the quality of the variables on the factor map. The PCs that are closer to the origin do a worse job of variables representation. Loadings were computed to provide an accurate measurement of the variables' contribution to the principal components (Appendix B, Figure 18). The absolute value of loadings indicates the strength of feature relationship with a principal component. The larger the loading - the stronger the relationship. The results of factor map are consistent with the correlation plot provided in the Figure 2. Variables with high positive correlations all point in the same direction and cover the same component.

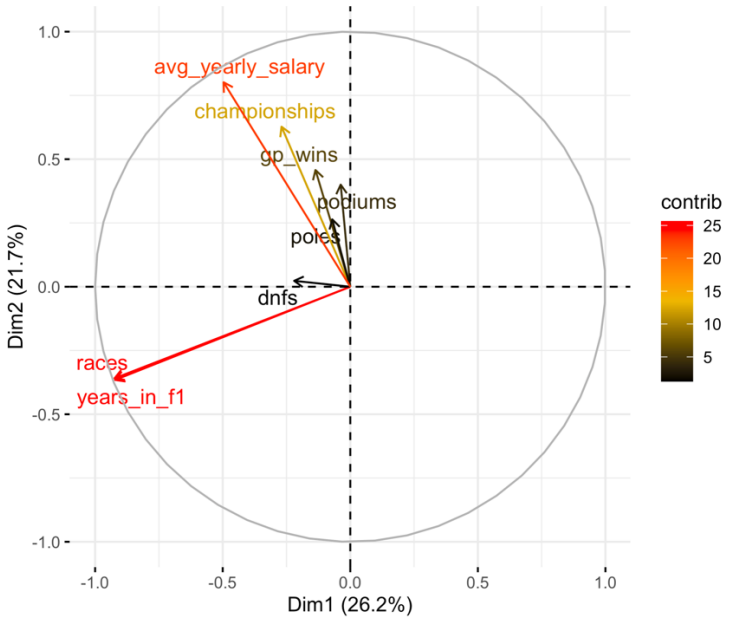

Figure 5. Features contribution to the principal components (PC1 and PC2)

The first principal component has a strong negative relationship with a number of races and years spent in Formula 1 and quite a strong negative relationship with the average yearly salary of the sportsman. However, the average yearly salary together with number of championship titles in the player's career are mainly represented by the second PC. The second principal component has strong relationships with almost all the features except 'dnfs' (the number of times the player did not finish the race). It is another evidence that the variables in the dataset shall be represented by more than two dimensions.

Coloring the observations projected to the PC1 and PC2 by player's country, player's team and player's favorite circuit, we cannot observe that they form any groups, which is quite logical and means that all the categorical features are associated with full range of the numeric ones (Appendix B, Figure 19).

2.2 K-MEANS CLUSTERING

The K-means Clustering technique is the first one we are going to apply. The goal of this vector quantization technique is to divide an input set of n observations into k clusters, with each observation being assigned to the cluster that has the closest mean among the clusters. This approach may be used to clusters of all sizes and forms, is straightforward to implement, and ensures convergence [11]. K-means is a numerical and iterative method, while Hierarchical Clustering also allows the usage of categorical data. Hence, we will perform K-mean on the numeric features of our dataset. Finding the number of clusters, k, is the first step in using the K-means clustering algorithm. We assess k using two methods: the

elbow method and the average silhouette method. The Elbow method is based on minimizing the within sums of squares (inter-cluster variance). We select a number of clusters such that including a new cluster does not significantly enhance the overall WSS. The average silhouette approach evaluates how well each data point fits into the cluster. The number of clusters k that maximizes the average silhouette score is the best one. Figure 6 shows the optimal number of clusters provided by the elbow and silhouette methods.

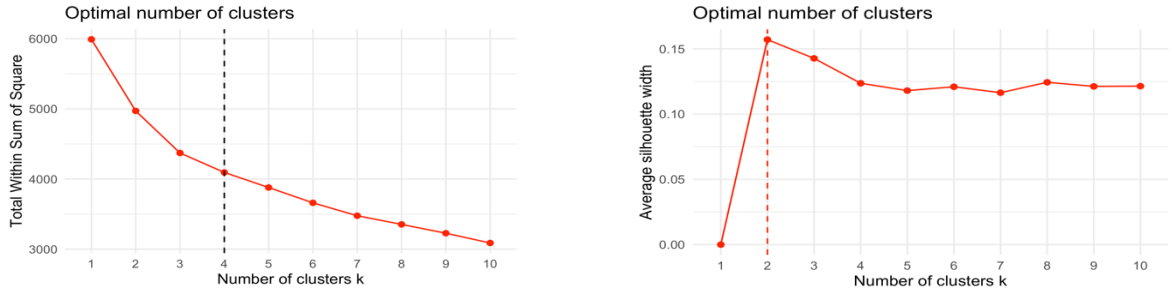

Figure 6. Optimal number of clusters for K-means clustering defined with an elbow (on the left) and silhouette (on the right) methods.

After considering both of the graphs, we select k=4 as the number of clusters. We run the analysis and plot the results using PC1 and PC2 (Figure 7). The obtained clusters are overlapping.

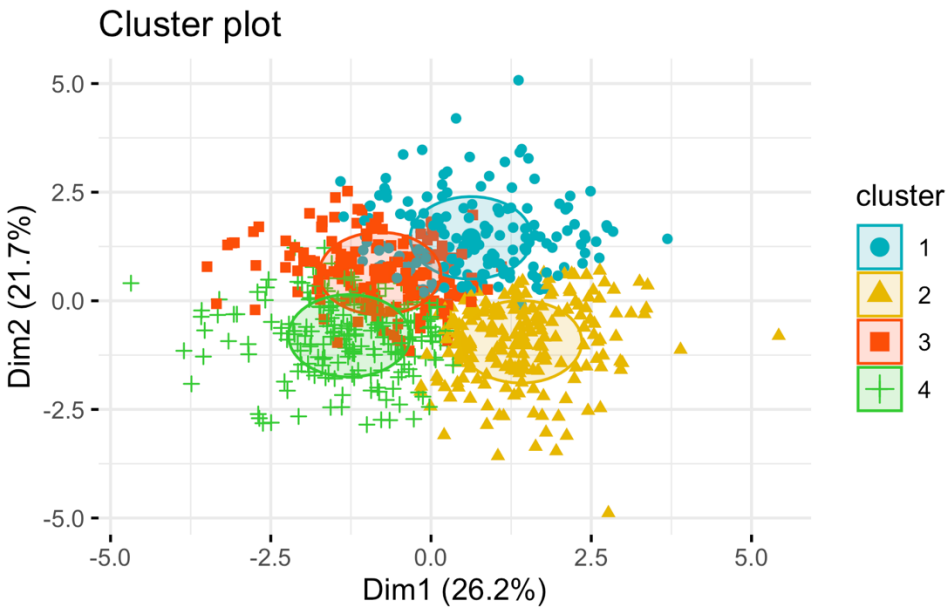

Figure 7. The result of K-Means clustering for the Formula 1 players. K = 4

It seems that the following clusters can be explained based on the features that contribute most to PC1 and PC2. However, we cannot specify it explicitly from the graph. To check this out, we plot the presence of each cluster among four

features: number of podiums for each player, number of races for each player, average yearly salary and years spent in F1 career (Appendix C, Figure 20).

Cluster 1 is rather interesting because it includes players with little (or relatively little) experience in Formula 1, an above-average yearly salary and a below-average number of races. Whereas cluster 2 identifies players with low scores in all groups. Clusters 3 and 4 show no clear trends and strongly overlap across all indicators.

Although the segmentation of players by quantitative parameters is insightful to a certain point, K-Means clustering cannot answer the question of whether there are patterns of cluster formation around categorical variables. For this purpose, we will use Hierarchical clustering.

2.3 HIERARCHICAL CLUSTERING

Hierarchical clustering is a technique that groups data points based on how close or similar they are to one another. It can be applied using a non-agglomerative or agglomerative approach. We use agglomerative clustering in our analysis. This approach considers the algorithm that starts from each data point as a separate cluster and moves forward until every point is in the same cluster [12]. Considering that for hierarchical clustering we include also categorical variables from our dataset, we rely on Gower's dissimilarity measure. For the linkage method we choose a complete one, so the distance between two groups is computed as the maximum distance between two different observations in the two different groups. The results of the clustering are presented using both normal and radial dendrograms (Appendix D, Figure 21). However, they are not informative due to the high number of observations.

We cut the dendrogram using the same number of clusters as in the K-Means Clustering method (k=4). The result still shows the clusters of players with the most similar performance metrics. However, our initial goal for Hierarchical clustering was to test what insight the presence of categorical variables adds to the analysis. To see how the categorical variables were distributed across clusters, we created a graph showing all levels of these features and the percentage of their presence in each cluster (Figure 8).

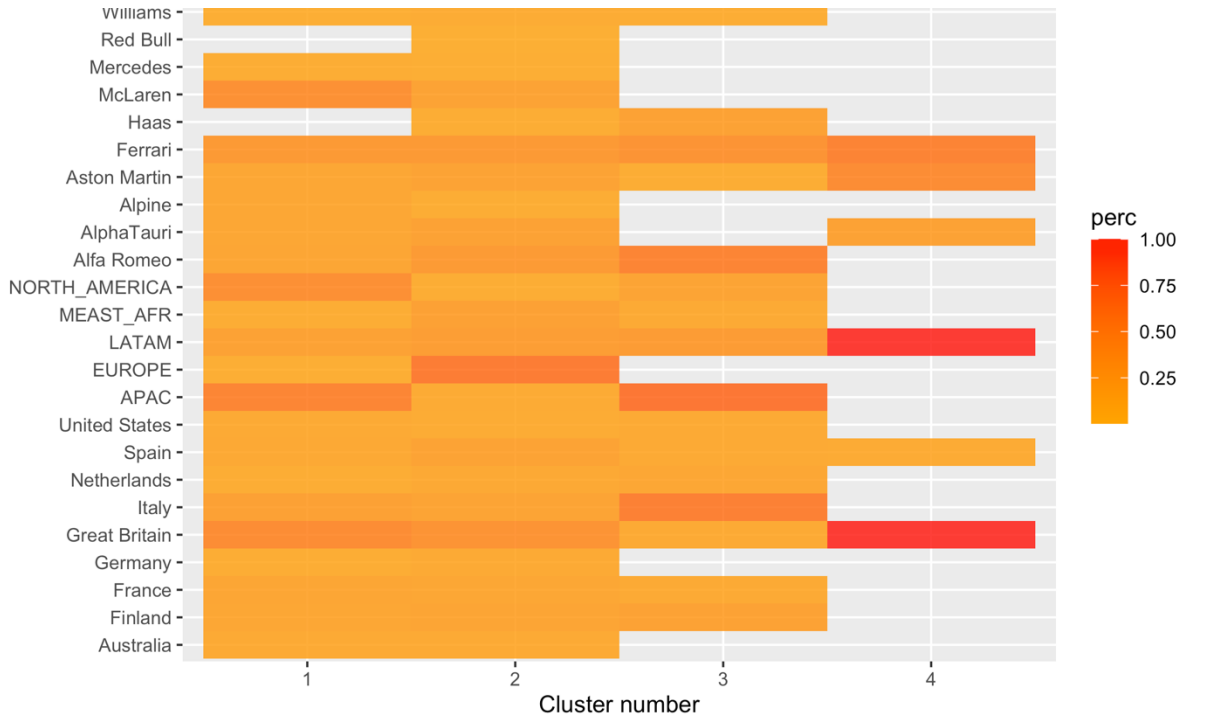

Figure 8. Distribution of characteristics across clusters

Again, it is impossible to single out any striking pattern for attributing this or that indicator to a particular cluster, but we can note that cluster number 4 is the most delineated, even though all of its characteristics are actively present in all other clusters.

The results of K-Means and Hierarchical clustering are consistent enough: both techniques demonstrated that four clusters are the appropriate number. On the one hand, Hierarchical clustering relies on the graphical representation and is not suitable for large datasets even though the number of clusters does not need to be predetermined and the results are repeatable. On the other hand, the main difficulties with K-means clustering are that the number of clusters must be predetermined, the results depend on where the initial centroids are placed, and the algorithm is sensitive to outliers.

4 SUPERVISED LEARNING: REGRESSION

As the main task for the supervised statistical learning part, we chose to predict and infer the average annual salary of Formula 1 players based on the available indicators. We use LASSO regression as a model for inference to select the features, after which we also build a linear regression for a more coherent interpretation of the model. We use the Random Forest method for average annual salary prediction.

Because the cleaned dataset contains only 678 entries, our regression models may overfit because they are trained on a small number of entities. Therefore, we use models with a small number of parameters and perform feature selection to improve the prediction power of the models. Since the target variable is blurred (due to large estimation error and privacy-preserving techniques), we expect the possible presence of anomalies in the data distribution.

4.1 LASSO REGRESSION

When there is multicollinearity in the data, we can use LASSO regression to fit a regression model. In addition to least squares regression, which seeks to minimize the sum of squared residuals (RSS), LASSO tries to minimize the following expression:

$$Cost\ (W) = RSS(W) + \lambda * (sum\ of\ absolute\ value\ of\ weights)$$
$$= \sum_{i=1}^{N} \left\{ y_i - \sum_{j=0}^{M} w_j\, x_{ij} \right\}^2 + \lambda \sum_{j=0}^{M} |w_j| \qquad (1)$$

This second term in the equation is known as a shrinkage penalty. In LASSO regression, we select a value for λ that yields the lowest possible test MSE [14]. We perform Lasso regression in R using the 'glmnet' package. Figure 9 reflects the performance of the regularization term (L1 norm) for LASSO. So as lambda (the weight given to the regularization term) approaches zero, the loss function of the model approaches the OLS.

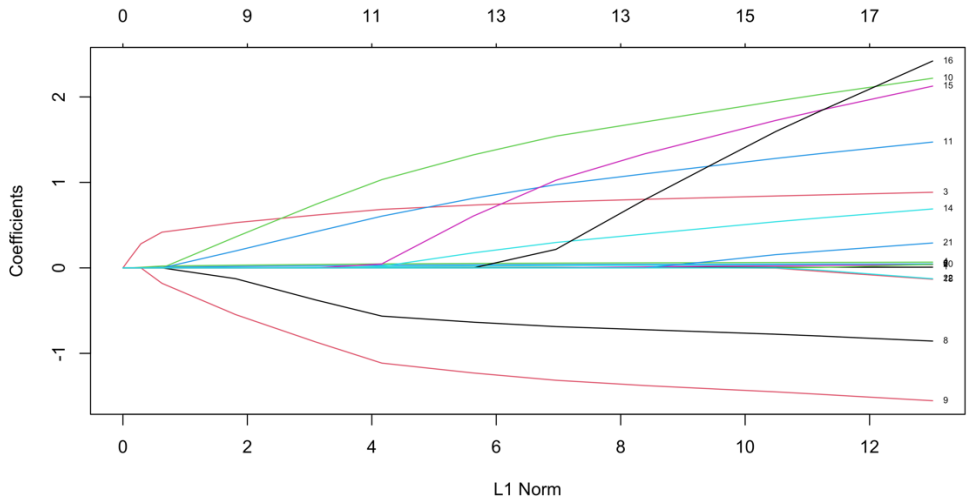

Figure 9. Regularization term graph for LASSO

When we have a small L1 norm, the model results in a lot of regularization and consequently returns an 'empty' model, and as we increase the L1 norm, variables

are entering the model as their coefficients take non-zero values. From the Figure 9 the first feature to take non-zero values is a feature number 3 (number of championship titles), which appears to be the most significant.

Next step is to calculate the optimal value of lambda to use. To do that, we perform a 10-fold cross-validation and identify the lambda that yields the lowest test MSE. According to our calculations, the lambda parameter set for the model is equal to 0.009, which makes LASSO regression very close to OLS regression. Figure 10 shows MSE depending on the lambda values.

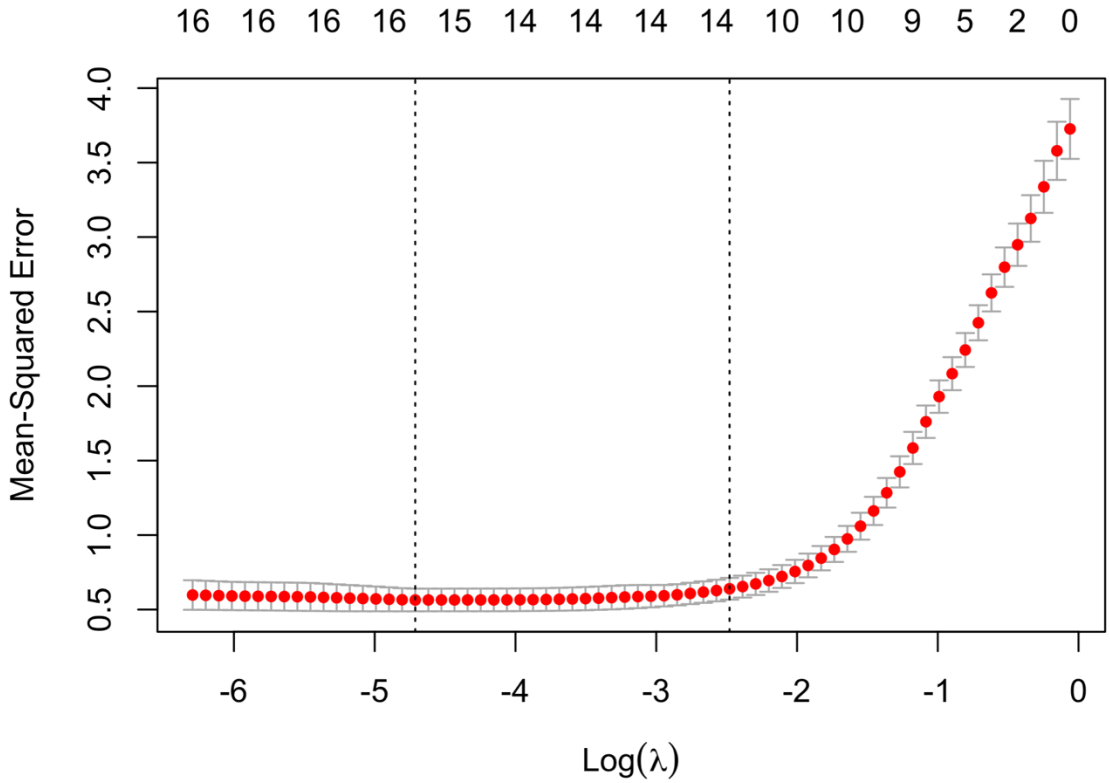

Figure 10. MSE by lambda value

Finally, we may analyze the final model obtained by the optimal lambda value. The R-square of the model is 0.872358 and coefficients provided are the shown in Figure 11. No coefficients are given for the predictors 'mc_laren', 'apac' and 'europe' because the LASSO regression shrunk these coefficients to zero. This means were completely dropped from the model because they were not influential enough. Interestingly, among the quantitative indicators, the strongest is the number of championship titles. According to the model, each championship title in a player's career adds $0.88 million to his average annual salary.

(Intercept)	0.241404044	alpha_tauri	-2.284194146
races	0.008996112	alpine	1.303760819
championships	0.880457811	aston_martin	0.732672462
gp_wins	0.069856537	ferrari	-0.716627559
podiums	0.049499427	mc_laren	.
poles	0.045394672	apac	.
dnfs	0.019194151	europe	.
alfa_romeo	-1.485367367	latam	0.119428046
		meast_afr	0.292508845

Figure 11. Model Coefficients provided by LASSO regression

'Alpha_tauri' team has the strongest negative effect among the dummy variables. If a player is a member of that car brand's team, his base annual earnings (characterized by an intercept) are reduced by $2.28 million.

However, we should be cautious about the interpretation of the proposed model, because during the data preprocessing stage we noticed that there is significant correlation between the dummies of the team and circuit variables, which means that there may still exist multicollinearity caused by dummy variables trap. To check this, we use linear regression.

4.2 LINEAR REGRESSION AND INTERPRETATION OF THE MODEL

Before fitting the linear regression model on all variables, we examine again the distribution and the pairwise correlations between the variables in the cleaned dataset (Appendix E , Figure 22). Then we build a linear regression model using all the parameters. The exceptions among the parameters (the same exceptions were done for the LASSO regression) were the Red Bull, Haas, Mercedes and Williams teams. These variables were excluded from the analysis because they represent unbalanced variables. The "North America" variable of the 'favorite circuit' group was also eliminated to reduce the number of features associated with 'favorite circuit' to n-1 levels. The results of the linear regression with the major part of the features are shown in Figure 12. They are quite consistent with the LASSO model, except this time the coefficient for the 'alpine' brand is not significant.

To test for multicollinearity, we calculate the variance inflation factor (VIF). And then compare the square root of it for each indicator to 2. If the square root of VIF is greater than two, there is multicollinearity in the model. As we see from our calculations, despite rather coherent regression results, multicollinearity in the model still exists. This indicates that the coefficients of the model cannot be trusted. To reevaluate the model, we need to choose a different subset of features.

```
               Estimate Std. Error t value             Pr(>|t|)
(Intercept)   1.3902735  0.3263287   4.260              0.0000234 ***
races         0.0092228  0.0006342  14.543 < 0.0000000000000002 ***
championships 0.8968855  0.0211402  42.426 < 0.0000000000000002 ***
gp_wins       0.0700381  0.0022668  30.897 < 0.0000000000000002 ***
podiums       0.0500767  0.0019974  25.070 < 0.0000000000000002 ***
poles         0.0464648  0.0023123  20.095 < 0.0000000000000002 ***
dnfs          0.0209095  0.0101844   2.053              0.040455 *
alfa_romeo   -2.8886647  0.1422083 -20.313 < 0.0000000000000002 ***
alpha_tauri  -3.6907353  0.1490692 -24.759 < 0.0000000000000002 ***
aston_martin -0.6187678  0.1498373  -4.130              0.0000410 ***
ferrari      -2.1073655  0.1394532 -15.112 < 0.0000000000000002 ***
mc_laren     -1.3790648  0.1426576  -9.667 < 0.0000000000000002 ***
latam         0.1331266  0.0640691   2.078              0.038107 *
meast_afr     0.3135398  0.0861949   3.638              0.000297 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.6651 on 662 degrees of freedom
Multiple R-squared:  0.888,     Adjusted R-squared:  0.8858
F-statistic: 403.8 on 13 and 662 DF,  p-value: < 0.00000000000000022
```

Figure 12. Linear Regression results (all features)

```
> vif(step.model)
      races championships       gp_wins       podiums         poles
   1.046358      1.018898      1.020250      1.009952      1.018895
      dnfs    alfa_romeo  alpha_tauri aston_martin       ferrari
   1.039902      5.181650      3.808501      3.734090      6.063249
  mc_laren         latam     meast_afr
   5.161352      1.051756      1.054089
> sqrt(vif(mod)) > 2 # MULTICOLLINEARITY!
      races championships       gp_wins       podiums         poles
      FALSE         FALSE         FALSE         FALSE         FALSE
       dnfs    alfa_romeo  alpha_tauri        alpine aston_martin
      FALSE          TRUE          TRUE          TRUE          TRUE
    ferrari     mc_laren          apac        europe         latam
       TRUE          TRUE         FALSE         FALSE         FALSE
  meast_afr
      FALSE
```

Figure 13. Variance inflation factor (all features)

The class including the team names is quite unbalanced. We exclude the least-represented team names from further analysis, leaving only the most-represented ones. Thus, to estimate the new regression model, from the team group we use only the dummy variables McLaren, Ferrari, and Alfa Romeo, which account for more than 66% of the all players. An alternative solution would be to resample the data to balance the groups, but this seems challenging due to the small size of the initial dataset. The results of the adjusted regression are shown in Figure 14.

```
Coefficients:
               Estimate Std. Error t value             Pr(>|t|)
(Intercept)   -0.185671   0.456199  -0.407              0.68414
races          0.008281   0.001026   8.072  0.00000000000000326 ***
championships  0.873234   0.034685  25.176 < 0.0000000000000002 ***
gp_wins        0.074352   0.003715  20.012 < 0.0000000000000002 ***
podiums        0.050028   0.003288  15.215 < 0.0000000000000002 ***
poles          0.043490   0.003792  11.470 < 0.0000000000000002 ***
alfa_romeo    -0.988714   0.120822  -8.183  0.00000000000000141 ***
ferrari       -0.232052   0.110915  -2.092              0.03680 *
mc_laren       0.501900   0.120567   4.163  0.00003557468554005 ***
apac           0.293735   0.109846   2.674              0.00768 **
latam          0.301961   0.110586   2.731              0.00649 **
meast_afr      0.379608   0.145475   2.609              0.00927 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 1.092 on 664 degrees of freedom
Multiple R-squared:  0.697,     Adjusted R-squared:  0.692
F-statistic: 138.9 on 11 and 664 DF,  p-value: < 0.00000000000000022
```

Figure 14. Linear Regression results (adjusted features)

Despite the negative multicollinearity test for the second model, we lost significantly in its explanatory power as the adjusted R-squared dropped from 0.8858 in the first model to 0.692 in the second one. Interestingly, the intercept

is not statistically significant in the new model, which is logical enough, because if the player is not racing, it is strange to expect the payroll to be different from zero. Nevertheless, the model still very clearly describes relationships of indicators in the luxury motorsport industry and can be used as a kind of benchmark for assessing the success of Formula 1 players.

As a verification step, we estimate the behavior of the residuals for the second regression model. Residuals versus fitted values plot, Scale-Location plot, residuals versus leverage plot as well as normal Q-Q plot are shown in the Figure 15.

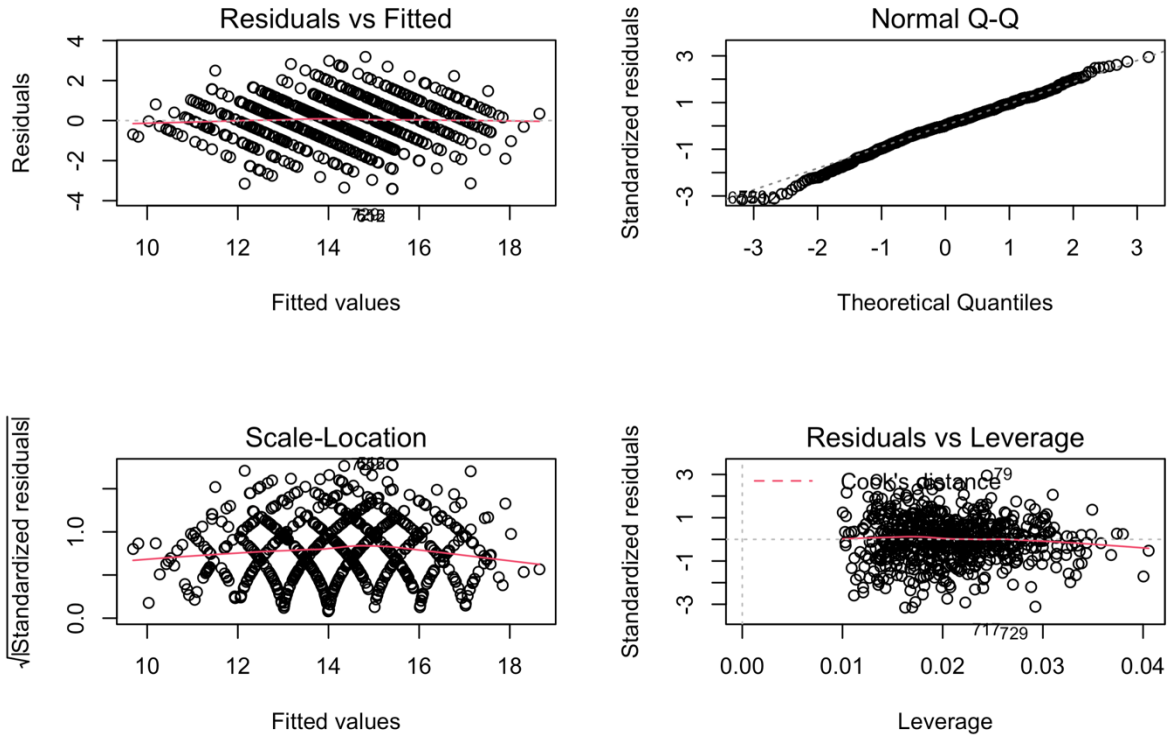

Figure 15. Residuals versus fitted values, Scale-Location, residuals versus leverage, normal Q-Q plots for the adjusted linear regression model

Despite the absence of clearly anomalous patterns, the normal Q-Q plot shows that the residuals at the extremes of the theoretical quantiles deviate quite strongly from the diagonal of the normal distribution [4]. This should definitely be taken into account. Next steps may include experimenting with logarithmic modeling to level out the residuals.

4.3 RANDOM FOREST REGRESSION

As the last model, which we consider more as a forecasting tool than as an inference tool, we took the Random Forest. The bootstrapping Random Forest algorithm combines ensemble learning methods with the Decision Tree framework to generate multiple randomly drawn decision trees from the data, then averages the results to produce a new result that frequently leads to strong predictions [15]. Random Forest is a low-biased, moderate-variance model that handles imbalanced data (overfitting problem is addressed by averaging the results over the trees). Furthermore, it is based on bootstrap resampling and training decision trees on the samples, which aids in the resolution of the problem of data scarcity.

In the first step, we apply the Random Forest Regression to a cleaned dataset of 678 players and 17 features. To evaluate the model, we divided the dataset into the training set (p = 0.7) and the testing set (p = 0.3). The percentages of variance explained by the algorithm are 70.46% on the training set and 63.33% on the test set with the default parameters (number of trees = 500, number of variables tried at each split = 5).

Next, using the feature importance plot, we investigate how much each indicator contributed to the model (Figure 16).

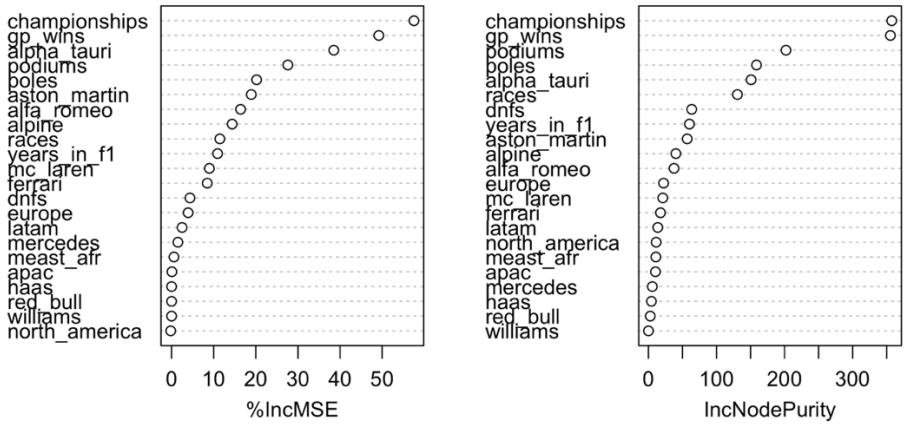

Figure 16. Importance of the indicators in Random Forest

The most important variables are: number of championship titles for each player and number of races won. The results are quite consistent with LASSO and linear models; however, the Random Forest model does not have good predictive ability in this case. In terms of prediction, multicollinearity in variables has no effect on the Random Forest. However, the feature importance score may be influenced: when correlated variables are included together in one split, their overall importance is reduced [15].

5 CONCLUSIONS

As a result of this work, we analyzed a dataset containing information about the career performance indicators of Formula One players, using supervised and unsupervised statistical learning techniques to understand what profiles the Formula One player community can be divided into, as well as what factors have the greatest impact on the average annual player's salary.

The results of K-means and Hierarchical Clustering are consistent and show that the dataset does not form any natural clusters. Nevertheless, in both cases, clusters are good at grouping players with similar career status. At the same time, both algorithms have significant drawbacks. K-Means is very sensitive to outliers, and Hierarchical Clustering does not allow to visualize its results even for relatively small datasets like ours.

In supervised statistical learning section, we pursued the goals of both prediction and data inference. The Random Forest Regression algorithm was chosen for prediction purposes. However, it showed rather mediocre results which can be caused by both the size of the dataset used and insufficient data quality.

For the question of data inference, we used the algorithm of feature selection using the LASSO Regression. And then we built linear regression models. Despite very good initial fitting results, subsequent inspection of the data revealed that the categorical variables were correlated between each other, leading to the problem of multicollinearity of the data and dummy variables trap. To avoid multicollinearity of the data, we made the subsequent choice of features, which significantly reduced the explanatory power of the proposed model.

All of the models designed fairly consistently describe the factors on which the average annual salary of a Formula 1 racer depends. Thus, one of the most influential factor is the number of championship titles, each of which increases the average annual salary of a player by almost $0.87 million (Figure 14). Fixed effects that depend on team selection are also important.

In the process of research, we also encountered various statistical challenges: unbalanced data, the problem of competent management of outliers, multicollinearity of data, non-normal distribution of residuals. Some of these problems we have solved locally, while others require a more detailed approach. In the case of the latter, we have only outlined the ways in which they can be solved.

REFERENCES

1. f1-salary-prediction. (2022, October 13). Kaggle. https://www.kaggle.com/code/niramay/f1-salary-prediction/data
2. GitHub - toUpperCase78/formula1-datasets: Datasets for Formula 1 World Championship. (n.d.). GitHub. Retrieved October 22, 2022, from https://github.com/toUpperCase78/formula1-datasets
3. Staff, S. (2022, October 23). How much are F1 drivers paid? 2022 salaries revealed. The Independent. https://www.independent.co.uk/f1/how-much-drivers-paid-2022-salary-b2211780.html
4. Khandelwal, A. (2021, September 6). Q-Q plot – Ensure Your ML Model is Based on the Right Distribution. Analytics Vidhya. https://www.analyticsvidhya.com/blog/2021/09/q-q-plot-ensure-your-ml-model-is-based-on-the-right-distributions/
5. Singh, D. (2019, November 12). Linear, Lasso, and Ridge Regression with R. Pluralsight. http://www.pluralsight.com/guides/linear-lasso-and-ridge-regression-with-r
6. Songhao, Wu (2020, May 19) Multicollinearity in Regression. Retrieved October 24, 2022, from https://towardsdatascience.com/multi-collinearity-in-regression-fe7a2c1467ea#:~:text=Multicollinearity%20happens%20when%20independent%20variables,variables%20into%20the%20regression%20model
7. Dummy Variable Trap. LearnDataSci. (n.d.). Retrieved October 26, 2022, from https://www.learndatasci.com/glossary/dummy-variable-trap/
8. Reusova, A (2018, April 1) Hierarchical Clustering on Categorical Data in R. Retrieved October 24, 2022, from https://towardsdatascience.com/hierarchical-clustering-on-categorical-data-in-r-a27e578f2995
9. F1 - The Official Home of Formula 1® Racing. (2022, March 18). Formula 1® - the Official F1® Website. https://www.formula1.com/en.html
10. Le Roux; B. and H. Rouanet (2004). Geometric Data Analysis, From Correspondence Analysis to Structured Data Analysis. Dordrecht. Kluwer: p.180.
11. MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. Vol. 1. University of California Press. pp. 281–297. MR 0214227. Zbl 0214.46201. Retrieved 2009-04-07.
12. Nielsen, Frank (2016). "8. Hierarchical Clustering". Introduction to HPC with MPI for Data Science. Springer. pp. 195–211. ISBN 978-3-319-21903-5.
13. Liu, Fei Tony; Ting, Kai Ming; Zhou, Zhi-Hua (December 2008). "Isolation Forest". 2008 Eighth IEEE International Conference on Data

Mining: 413–422. doi:10.1109/ICDM.2008.17. ISBN 978-0-7695-3502-9. S2CID 6505449.

14. Tibshirani, Robert (1996). "Regression Shrinkage and Selection via the lasso". Journal of the Royal Statistical Society. Series B (methodological). Wiley. 58 (1): 267–88. JSTOR 2346178.

15. Beheshti, N. (2022, March 2) Random Forest Regression. Retrieved October 20, 2022, from https://towardsdatascience.com/random-forest-regression-5f605132d19d

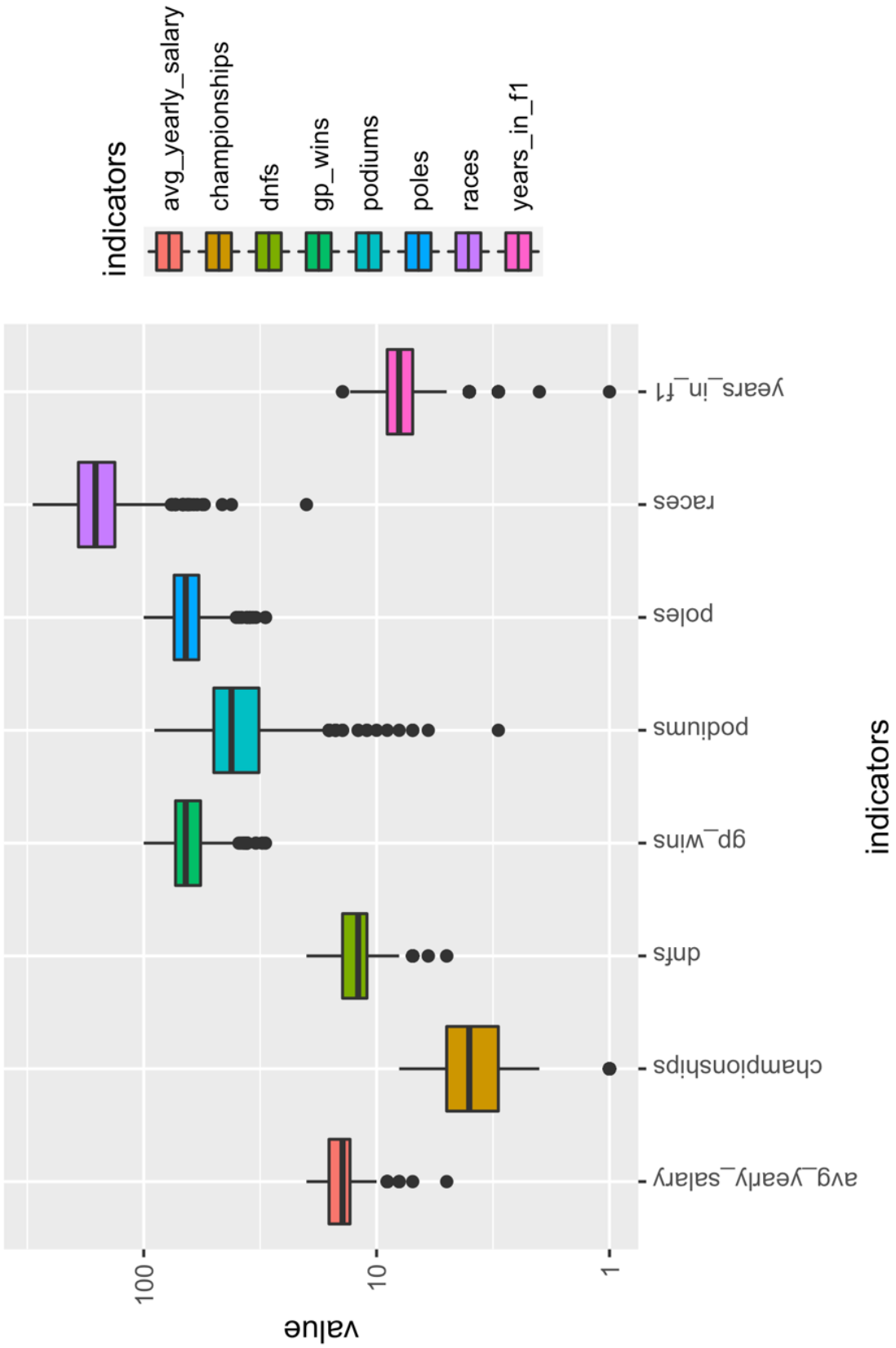

Figure 17: Boxplots for the numeric variables.
NB: the graph uses the y-log scale.

APPENDIX B

	PC1	PC2	PC3	PC4
races	-0.63819552	-0.27225760	-0.01559759	-0.02006993
years_in_f1	-0.63664002	-0.27716338	-0.01338186	-0.01476176
championships	-0.18613696	0.47581035	-0.11503518	-0.10974813
gp_wins	-0.09426305	0.34700509	0.55383797	-0.44505499
podiums	-0.02595256	0.30353922	0.08453401	0.71948754
poles	-0.04887155	0.20036694	-0.78930968	-0.07363667
dnfs	-0.15195814	0.01753696	0.22065889	0.51528613
avg_yearly_salary	-0.34308925	0.60799484	-0.02760335	-0.02561675
	PC5	PC6	PC7	PC8
races	0.10310419	0.02585313	0.08045897	0.707222518
years_in_f1	0.10328223	0.01642034	0.08231507	-0.706937294
championships	-0.13981858	-0.69354490	0.46164746	0.002766893
gp_wins	-0.03705641	0.48694485	0.35690207	-0.004894516
podiums	0.52167993	0.17142856	0.28432956	0.001046961
poles	-0.18069154	0.48891628	0.23940887	-0.005231653
dnfs	-0.80675072	0.10654048	0.01385108	0.001257880
avg_yearly_salary	0.04619297	0.03412361	-0.71267199	-0.003783130

Figure 18: PCA loadings

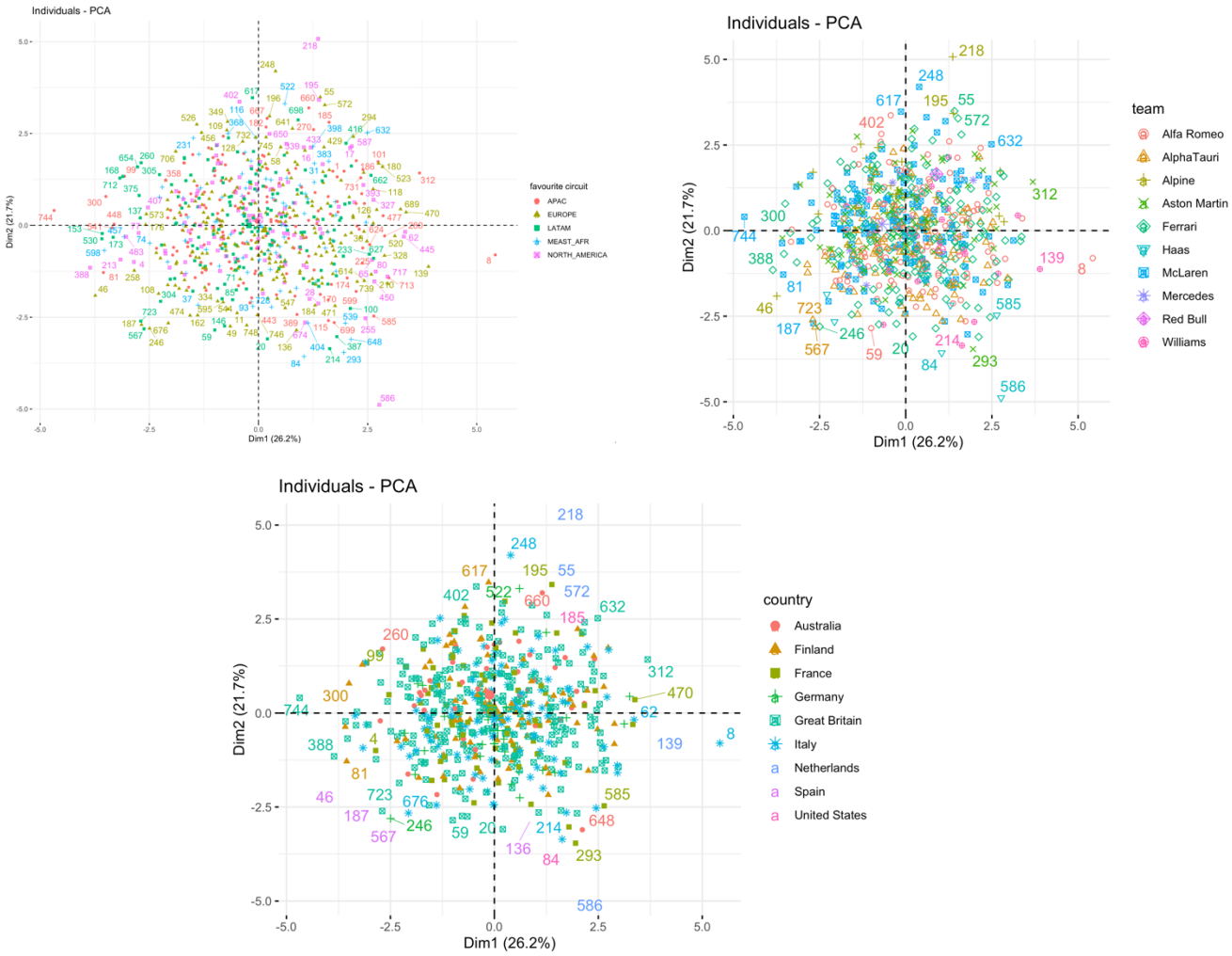

Figure 19: Observations projected to PC1 and PC2, colored by player's favorite circuit, by player's team and by player's country.

APPENDIX C

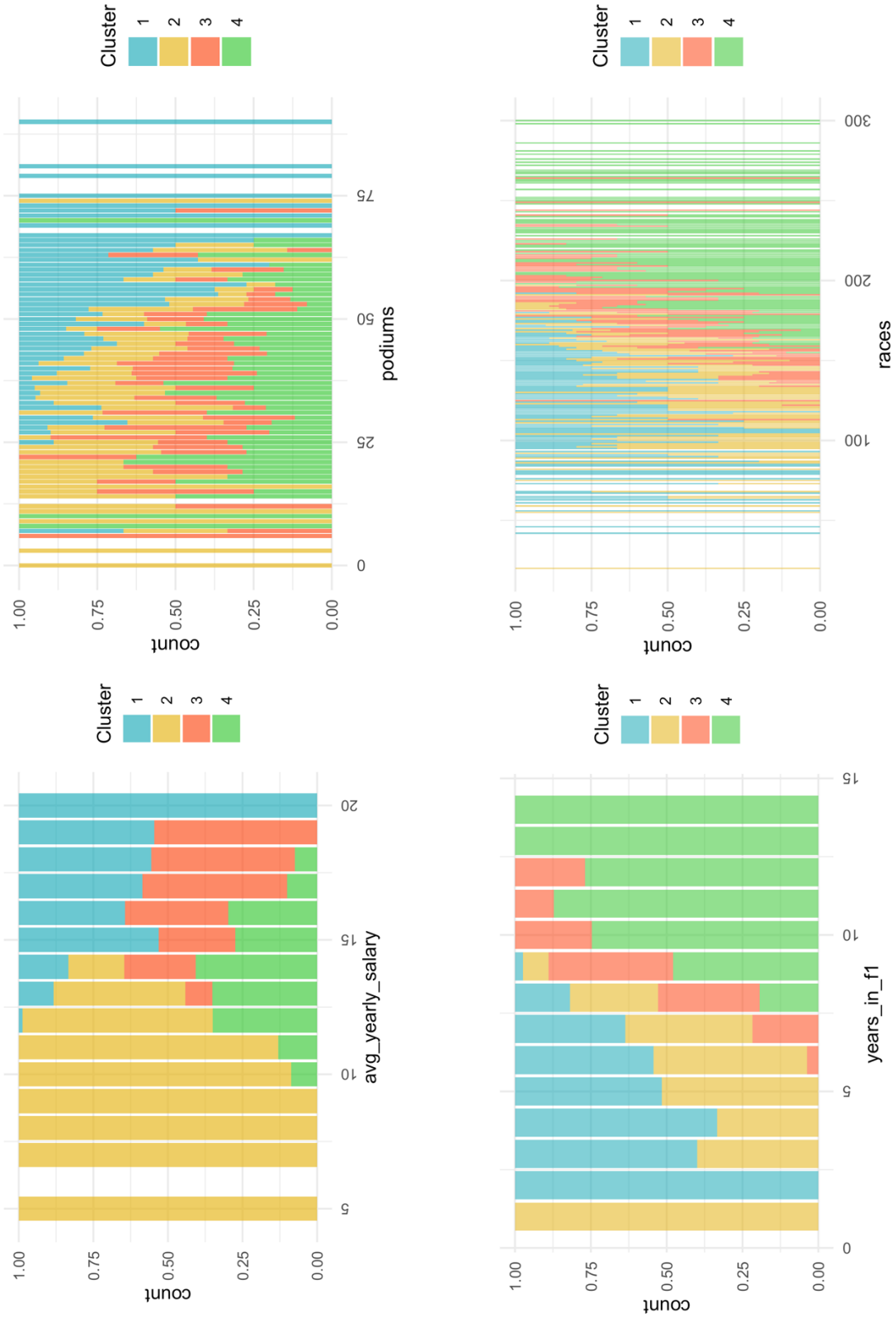

Figure 20: K-Means Clustering results (k=4) for the different features: number of podiums for each player, number of races for each player, average yearly salary and years spent in F1 career.

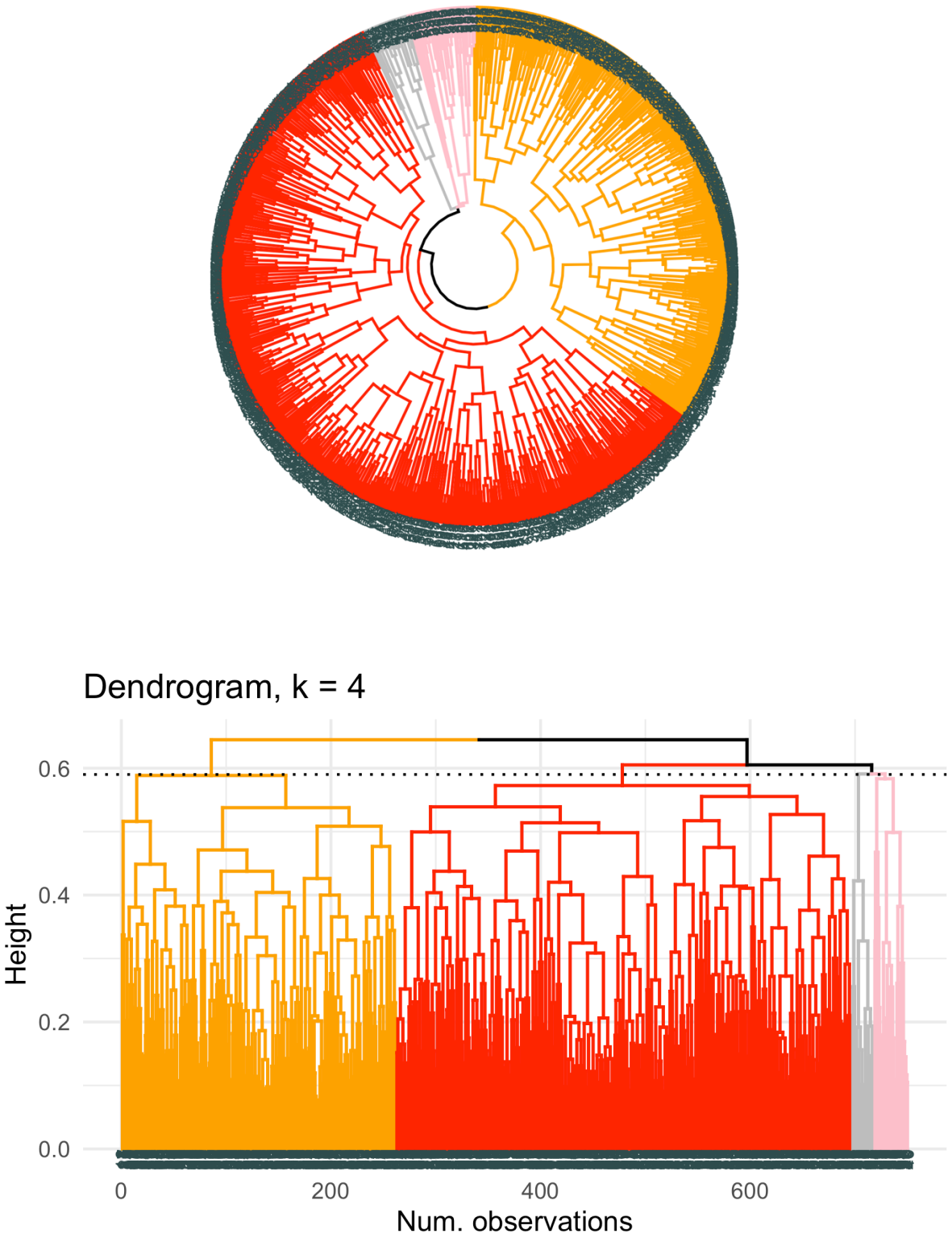

Figure 21: Dendrograms of Hierarchical Clustering (radial and normal forms) with all the F1 players split into 4 clusters. Ward linkage applied

APPENDIX E

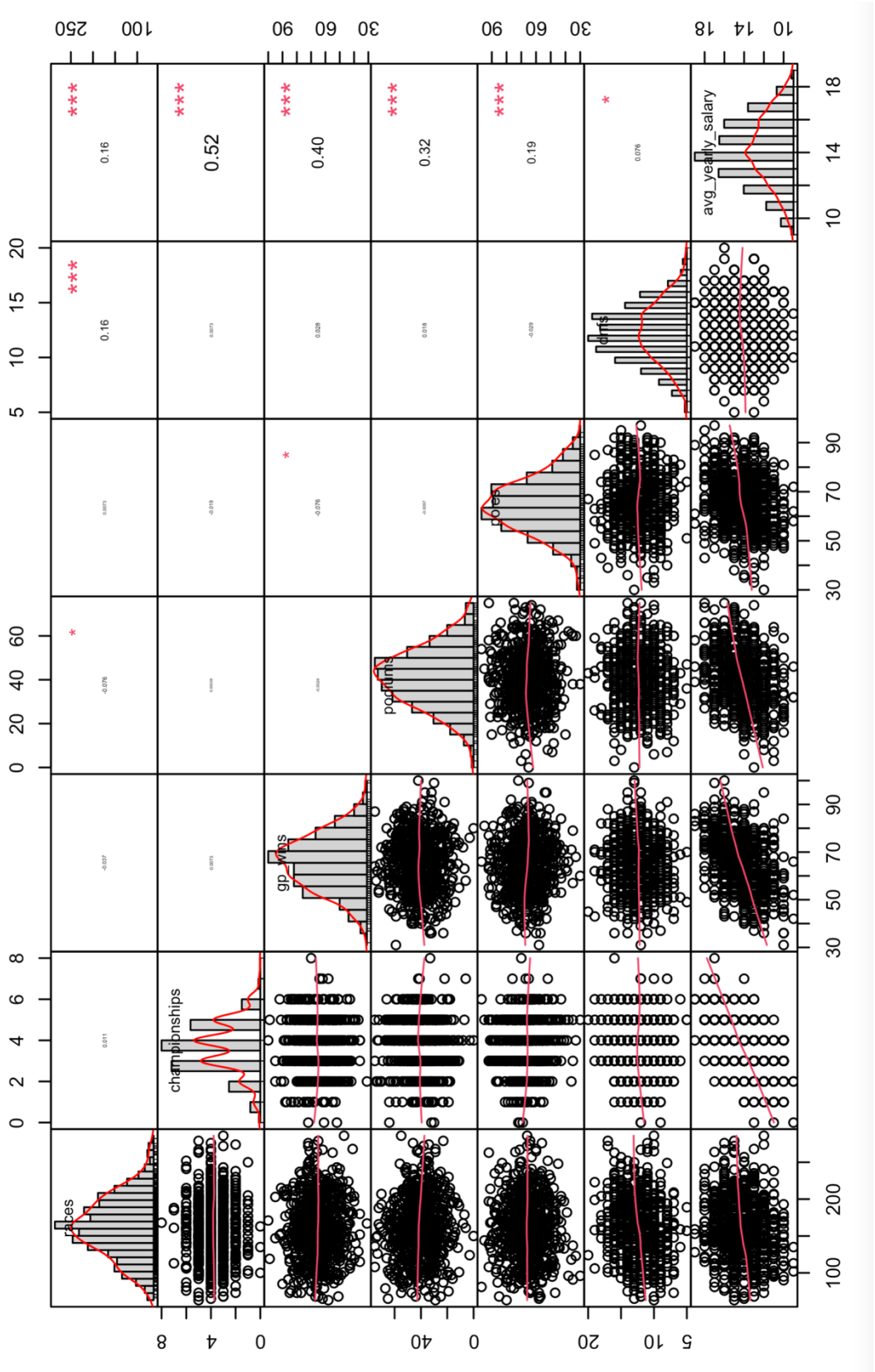

Figure 22: Distribution of the variables used for the linear regression model and their correlation plots