Final Report

Nick Mandarano and Patrick McHugh

4/26/2022

Introduction

To recap our proposal, we chose to do an NBA-related project and decided to fit a regression model for salary of NBA players, using statistics and other information about the players as covariates.

Our data includes many seasonal statistics for each player, including both simple stats and advanced metrics. We also have access to other personal information for each player, such as team, position, height, etc.

We are modeling based on the 2015-16 NBA season. We know salary is known before the season starts and statistics are created, so it is not a response variable in the traditional sense of a causal effect. We attempt to explore the relationship between salary and the covariates, and aim to prescribe a true mean function, and identify players who may be overperforming or underperforming their contract according to the 2015-16 market; i.e., putting up better or worse statistics than one might expect a player on their salary to do.

Data Exploration

Our data comes from four different datasets. We used three of Riguang Wen's datasets from figshare.com – `players cv`, `players salary`, and `players stat`. We also used a dataset called `NBA RS 2020-1950 Stats` uploaded to zenodo.org by Pablo Gomez and Sandra Giral.

In addition to player salary, the data available to us included statistics from each player for the season, including basic stats such as games played, points, and steals, as well as advanced stats such as Value Over Replacement Player (VORP), Defensive Box +/- (DBPM), and others. Other information included personal and demographic data related to the players, such as age, height, college attended, birthplace, and other things. A full list of variables can be found in the previously submitted EDA description.

We proceeded to filter out players who were below the league minimum in salary, as they were exclusively players signed to short term (i.e., 10-day) contracts with wildly volatile data. We felt that players on shorter than full season contracts would be worth creating a separate model for in another project.

From our EDA, we concluded that a log transformation of the response variable, salary, would be appropriate based on its skewed distribution.

We also noticed that many of the covariates had skewed distributions, or did not have linear marginal relationships with the response variable log(salary). We attempted to make these variables approximately normally distirbutied with an approximately linear marginal relationship with the response. We experimented with log, square root, and squaring transformations. In the end, we consider the following transformations in our model:

- VORP: A min/max normalization between 0 and 1 to eliminate negative values; then a log transformation
- OWS: A min/max normalization between 0 and 1 to eliminate negative values; then a log transformation

- DWS: Square root transformation
- PTS: Square root transformation
- FTr: Square root transformation
- BLK: Log transformation
- STL: Log transformation
- GS: Log transformation
- ORB: Log transformation

We use VORP as an example in the appendix. Figure A1 shows the distribution of VORP and its plot against log(salary) both pre- and post-transformation. Though not perfect, the transformed variable appears to be much more appropriate for a linear model than the raw variable.

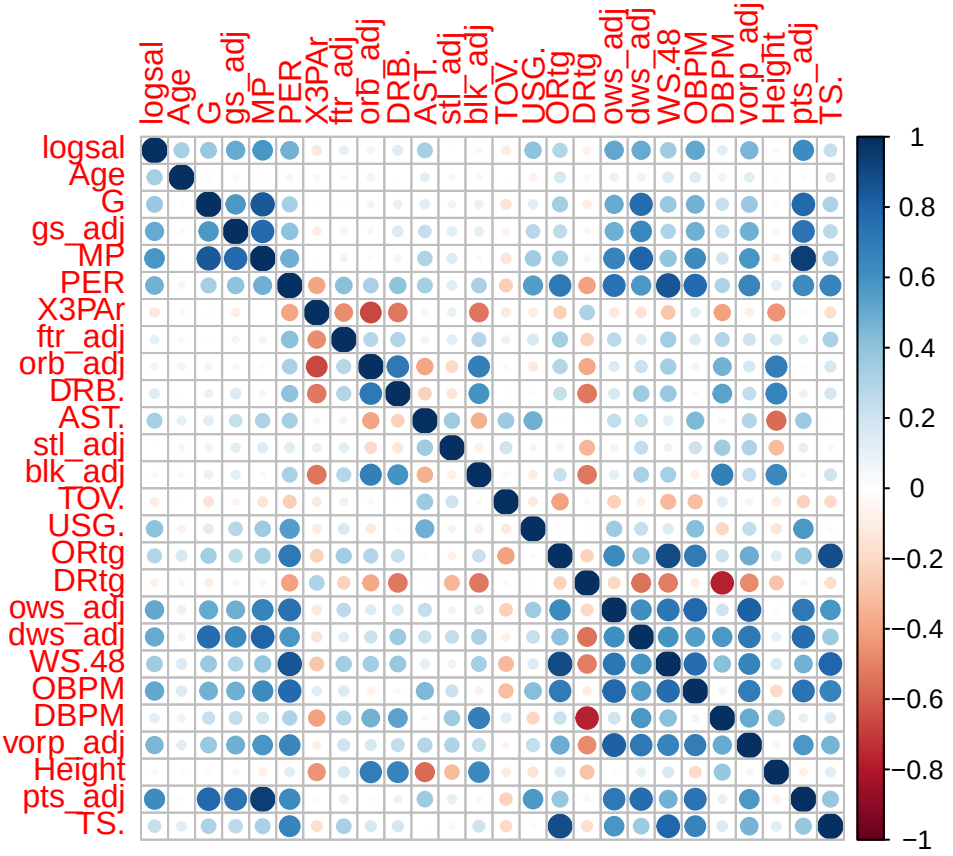
We can also see that Pos has some redundancy and maybe too much granularity. For example, "F-C" (forward/center) and "C-F" (center/forward) are treated as different positions by the model when they are functionally the same. We cleaned this up by grouping players into "Guards", "Wings", and "Bigs" according to Pos. Figure A2 shows this variable pre- and post-transformation. From the graph, it is unclear if there is a significant relationship between the refined position predictor Pos_cat and log(salary).

We modified some other categorical variables as well. Over several iterations of model building, we were able to reduce the Team variable to a simple flag, Multiteam. A player appearing with multiple teams over the course of the season had a strong relationship with salary.

Once our predictors are appropriately transformed, we considered all pairwise correlations between continuous covariates as an initial search for possibly collinearity, which we will discuss in more detail later.

Variable Selection

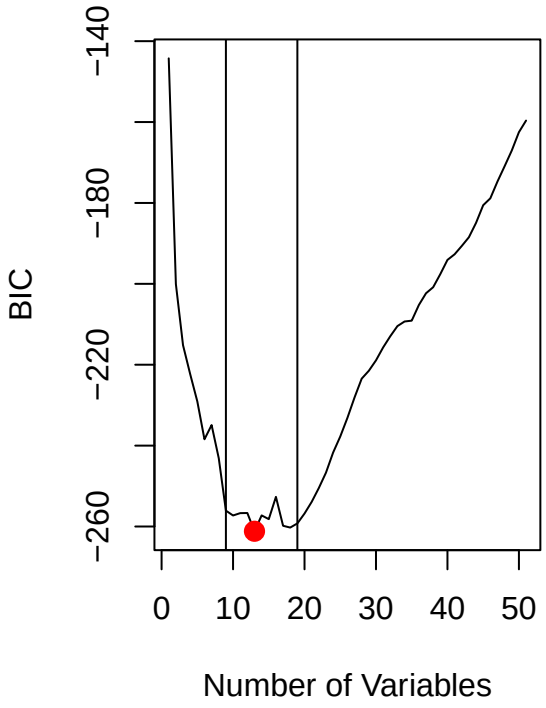
Three pairs of predictors with noticably large correlations according the the correlation matrix are further investigated. We learn that pts_adj and MP have a correlation coefficient of 0.947, ORtg and TS. have a correlation coefficient of 0.888, and finally WS.48 and PER have a correlation coefficient of 0.864. To avoid redundancy in the model, we'll remove one predictor in each of the three pairs from the model. The terms' variance inflation factors will guide the decision regarding which predictor to drop. Using $\text{GVIF}^{\frac{1}{2df}}$ will allow the GVIFs to be comparable across dimensions. Thus, we remove pts_adj ($8.77 > 6.25$), TS. ($4.64 > 4.29$), and WS.48 ($8.42 > 7.42$).

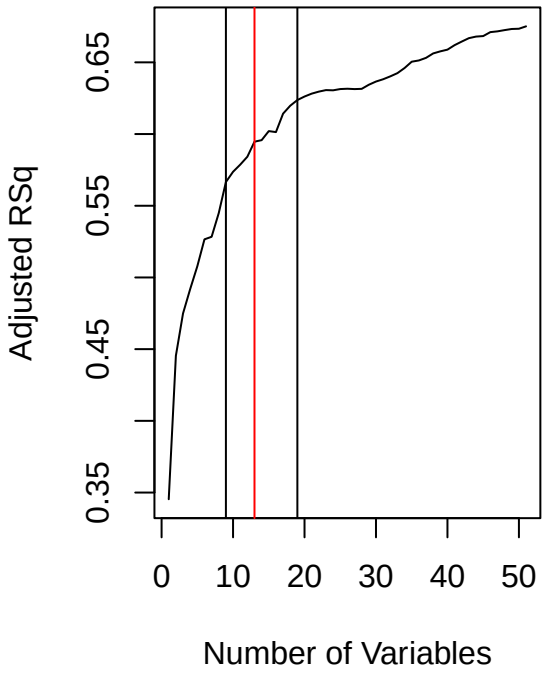


We will begin by looking at a full model with all remaining predictors and employ stepwise regression methods to find the best model. According to BIC, the stepwise regression model working in both directions returns MP, Age, USG., G and orb_adj as predictors. We'll call this Model A. The model using AIC in its variable selection process additionally returns AST., stl_adj, Pos_cat, Multiteam, and TOV.. We'll call this Model B.

Interactions of predictors may also be of interest in our model. Intuitively, it would make sense if the effect of offensive rebounds on a player's salary was dependent on the player's position. Therefore, we'll also consider models with interaction terms, but with 26 possible main effects, there are at least $26(25) = 650$ possible interaction terms. We'll use the regsubsets function in the leaps package to return the best models of each size, setting a conservative limit of the number of variables in the model to 50.

```
## Reordering variables and trying again:
```





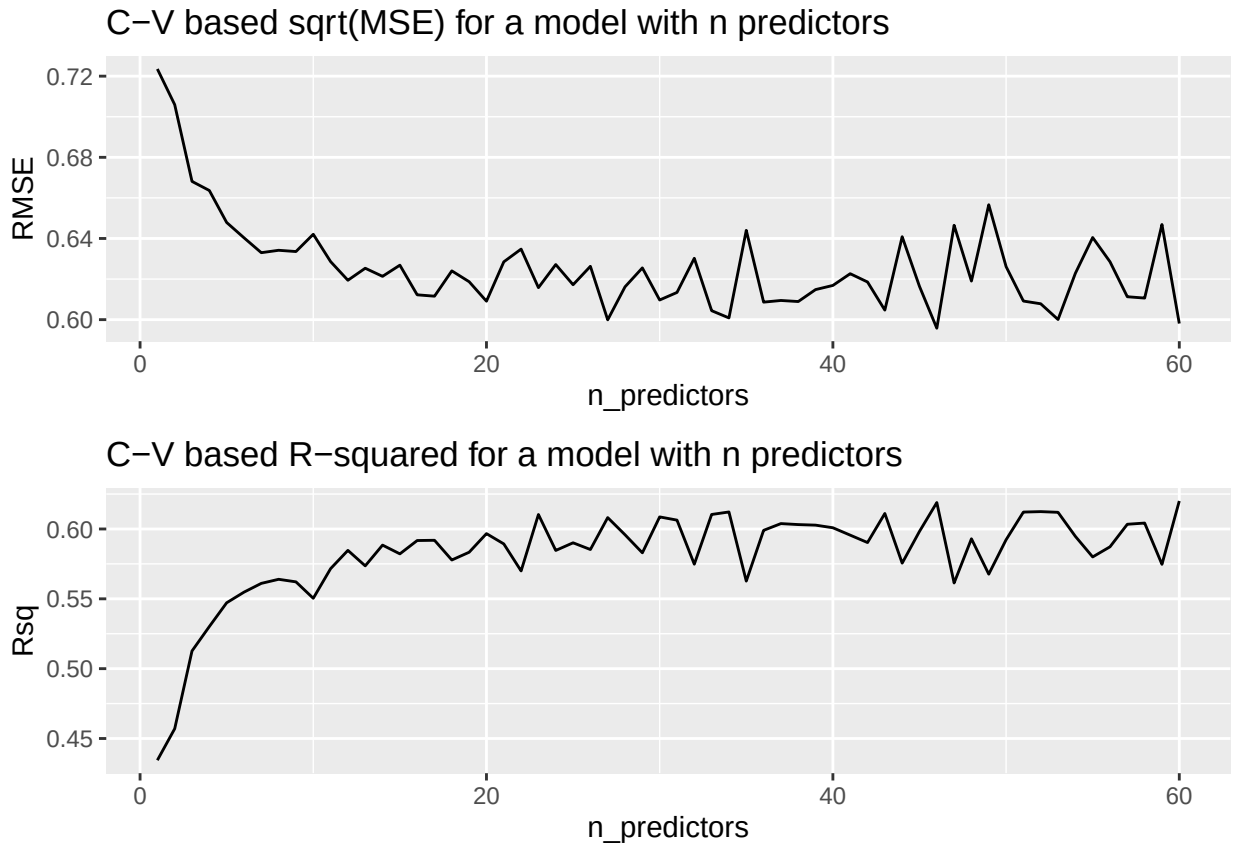
As we continued the model building process, we wanted to make sure we were avoiding overfitting. We created an algorithm where we used the regsubsets function to identify the n best predictors for all n from 1 to a large number, and included first level interaction effects. While some of these models included interactions without the corresponding main effects, we just used this as a rough guideline. We then looked at some of the evaluation metrics for these "best n predictors" models, such as R^2 and MSE, on k-folds cross-validated data.

```
train.control <- trainControl(method = "cv", number = 5)
datax = data %>%
    dplyr::select(-c(Pos_cat, International, Conference))
rsq = c()
rmse = c()
nmax = 60
regfit_full <- regsubsets(logsal ~ .^2, data = datax, nvmax = nmax, method = "backward")

for (i in 1:nmax) {
    cols = names(which(summary(regfit_full)$which[i, -1] == TRUE))
    predictors <- paste(cols, collapse = "+")
    form = as.formula(paste0("logsal ~", predictors))
    cv.mod = train(form, data = datax, method = "lm", trControl = train.control)
    rsq = c(rsq, cv.mod$results$Rsquared)
    rmse = c(rmse, cv.mod$results$RMSE)
}


plt_data = data.frame(n_predictors = 1:nmax, RMSE = rmse, Rsq = rsq)
```

```
rmse_plt = plt_data %>%
    ggplot(aes(x = n_predictors, y = RMSE)) + geom_line() + ggtitle("C-V based sqrt(MSE) for a model wi
rsq_plt = plt_data %>%
    ggplot(aes(x = n_predictors, y = Rsq)) + geom_line() + ggtitle("C-V based R-squared for a model wit
grid.arrange(rmse_plt, rsq_plt, nrow = 2)
```



As the number of predictors increased, the model metrics tended to flatten out. While it was good that the cross-validation based metrics weren't getting worse, the rate of improvement decreased heavily. Also, as more terms are added to the model, training set metrics such as R^2 and MSE continued to rise. The increasing gap between training metrics and cross-validation metrics concerned us, and guided us towards the simpler models. Also, with similar c-v based evaluation metrics, we prefer simpler models, due to interpretability.

Of these best models previously mentioned, we see that the model with the smallest BIC is the best model chosen with 13 variables. Though the Adjusted R^2 of a model does not necessarily have to increase as more variables are added, we see that in this case, the Adjusted R^2 is non-decreasing in the number of variables at least up to 50. Practically, a model with 50 variables is not ideal. There seems to be a point in the Adjusted R^2 plot in which the rate of change begins to flatten out that matches well with the point in the BIC plot where the BIC begins to rise again. This point is the best model chosen with 19 variables. Similarly, the best model chosen with 9 variables corresponds to an appropriate lower bound for the number of variables with both a satisfactory BIC and Adjusted R^2.

The best model chosen with 9 variables, which we'll call Model C, includes the following predictors:

- MP
- X3PAr

- ows_adj
- Age:DRtg
- G:ftr_adj
- blk_adj:TOV.
- ORtg:DBPM
- DRtg:vorp_adj
- dws_adj:ConferenceWest

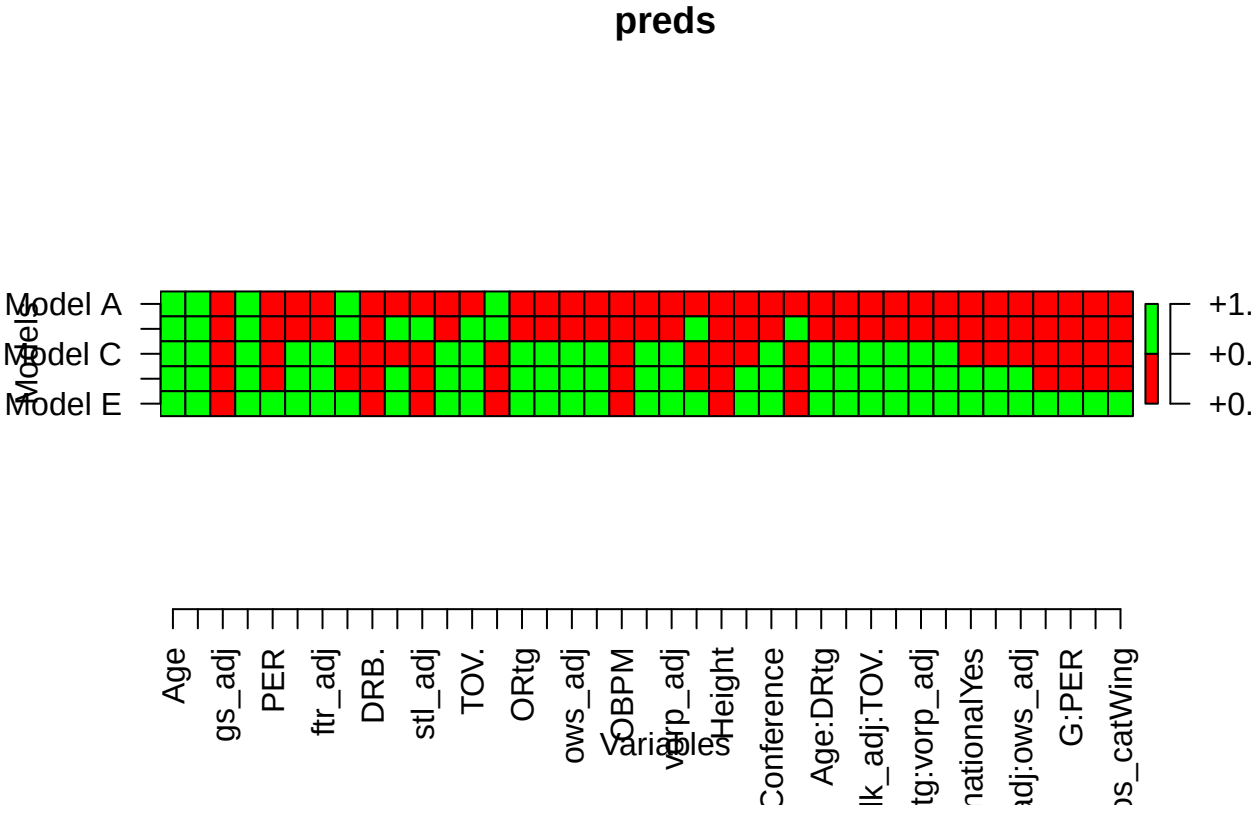
The best model chosen with 13 variables, Model D, includes the aforementioned 9 in addition to:

- Age
- Age:InternationalYes
- AST.:vorp_adj
- blk_adj:ows_adj

The best model chosen with 19 variables, Model E, includes the previously listed 13 as well as:

- ftr_adj
- DRtg
- Age:ows_adj
- G:PER
- ftr_adj:Pos_catWing
- orb_adj:Pos_catWing

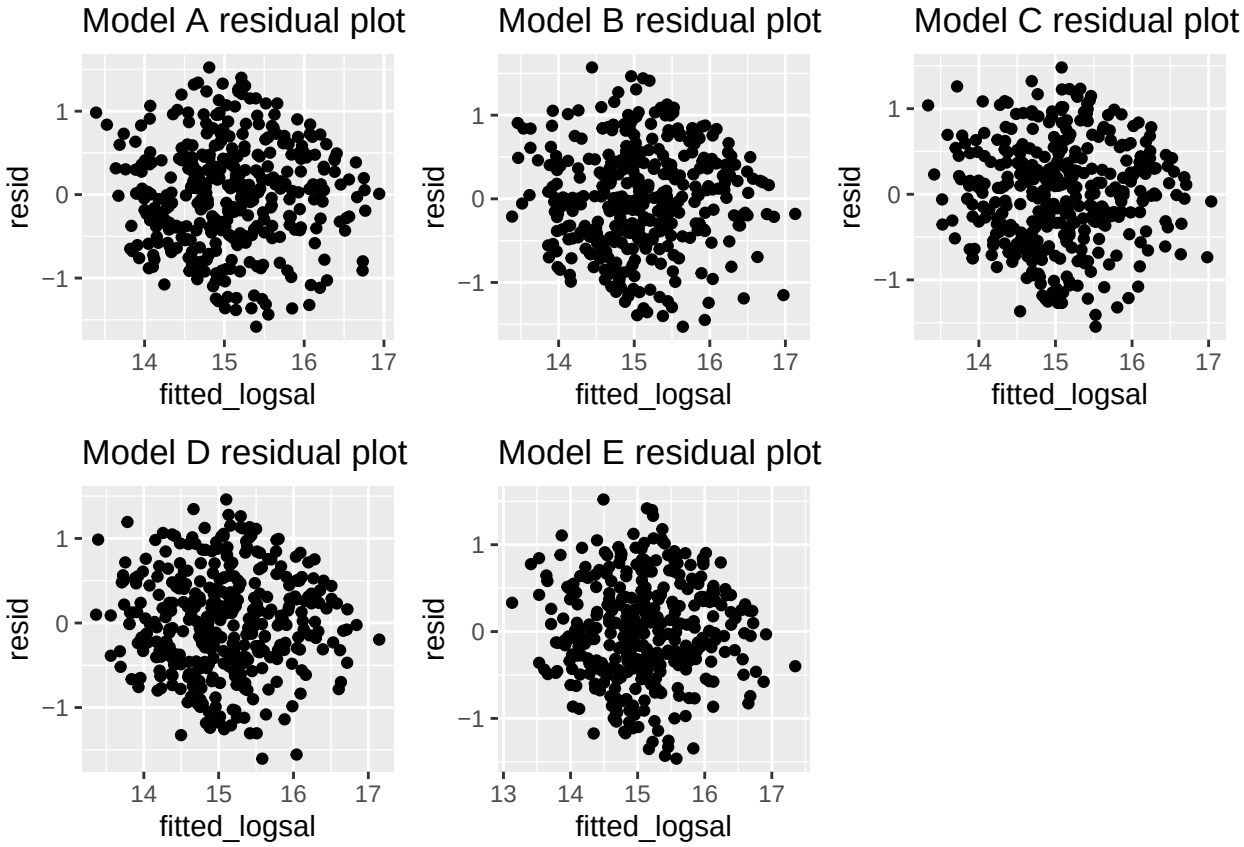
However, we would like to follow a rule of thumb that if a predictor is involved in an interaction term within a model, the main effect should also be included. Therefore, we'll add terms to each of the above models in order to suffice this rule. Thus, for the five models proposed so far, we examine the predictors involved.



Immediately we can see that all five proposed models use Age, G and MP as predictors. On the other hand, none of the models use gs_adj, DRB., OBPM or Height.

We'll fit each of the models and explore the residual plots.

Residual Analysis



The residual plots do not appear too problematic. However, a vague downward trend is apparent in each and the variance is not constant across all fitted values. Thus, these are not null plots. Fitting the models using weighted least squares may be of interest in the future. One persisting concern is the idea of overfitting the model. In order to evaluate whether any of these models may be victim to overfitting or underfitting, we'll perform 6-fold cross-validation on each model. The number of folds is chosen to be 6 because the data has 366 observations, so the folds will split evenly.

Model Selection

```
##            Model.Rsquared Model.AdjRsquared   CV.RMSE CV.Rsquared    CV.MAE
## Model A 0.561542825702464  0.55545314272611  0.638518    0.554957 0.5163634
## Model B 0.587068772936111 0.574237576614916 0.6318266   0.5665491 0.5120201
## Model C 0.587656442015736 0.562484306208557  0.642705   0.5575092 0.5264573
## Model D 0.592495638254362 0.561241616409564 0.6581126   0.5317355 0.5398083
## Model E 0.623956217060371 0.582808569079135 0.6645744   0.5348927 0.5403748
```

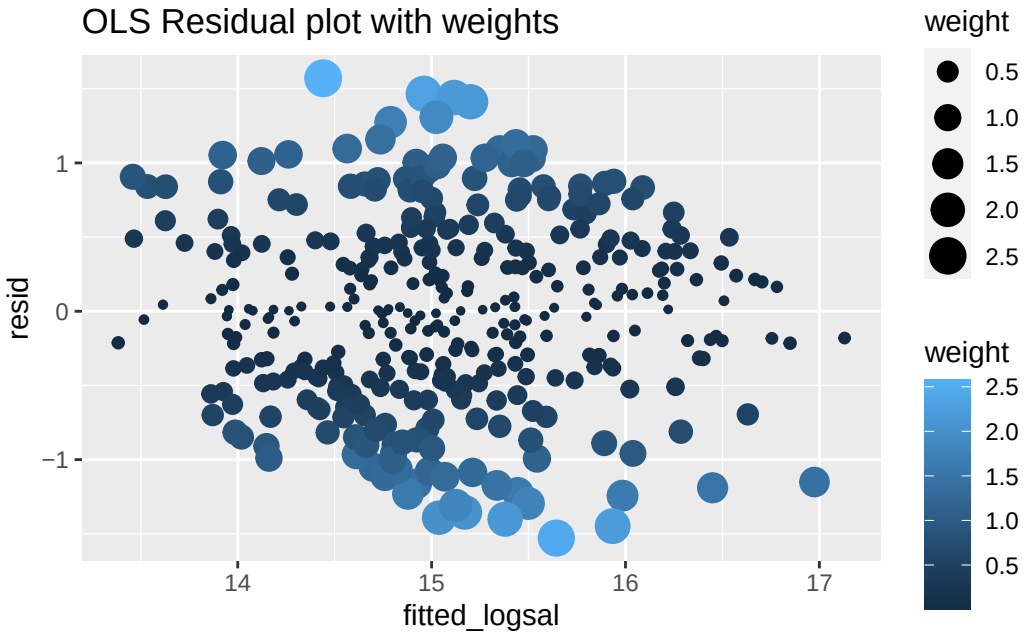
Here, Model.Rsquared is the R^2 reported by the model and Model.AdjRsquared is the Adjusted R^2 reported by the model. Meanwhile, CV.RMSE is the average root mean squared error of the model on unseen data

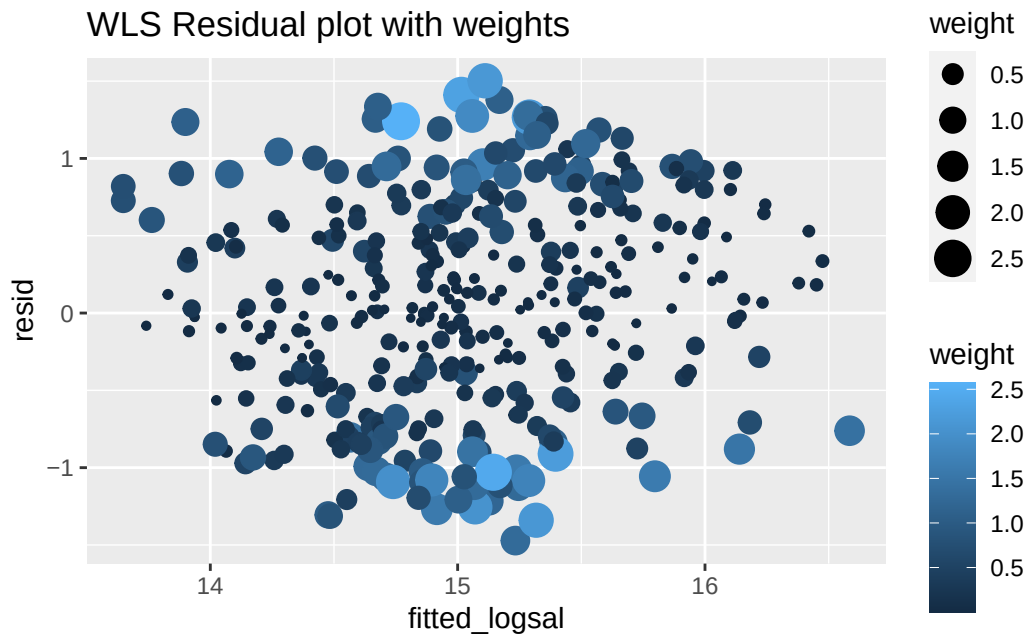
from the cross-validation, CV.Rsquared is the R^2 of the model on unseen data from the cross-validation, and CV.MAE is the mean absolute error of the model on unseen data from the cross-validation. Ideally, we're searching for higher values for Model.Rsquared, Model.AdjRsquared and CV.Rsquared with lower values for CV.RMSE and CV.MAE. However, too small of a difference between model accuracy and cross-validation accuracy may be an indication of overfitting, while too large of a difference may be a symptom of underfitting.

Considering Weighted Least Squares

A previously mentioned concern was the vague downward linear trend of the residuals. We will attempt to use weighted least squares to help resolve this. Our first attempt will involve plotting the standardized residuals against each of the predictors as well as the inverse of each of the predictors. This will help display if the variance in residuals is a function of any of the individual predictors.

No clear relationship between the residuals and any predictor is observed, so we instead use the HC3 method and compute the weights as a function of the OLS residuals and the leverages.






```
## OLS:
```

```
##   intercept      RMSE  Rsquared       MAE     RMSESD RsquaredSD      MAESD
## 1      TRUE 0.6347807 0.5681924 0.5134811 0.04560952  0.1059659 0.03880374
```

```
## WLS:
```

```
##   intercept      RMSE Rsquared       MAE     RMSESD RsquaredSD      MAESD
## 1      TRUE 0.7177479 0.448928 0.5889058 0.04701361 0.05170899 0.05281043
```

We notice that the residual plot for the WLS is marginally better. However, when evaluating each model over k-folds, the WLS performs poorly. We suspect this is probably a case of overfitting. The OLS model is better and it's not even close. Thus, we decided to table the WLS idea for now.

Discussion

Model Interpretation

In our model, four predictors are significant at the 99.9% confidence level: MP, Age, G and AST. The estimated coefficients for MP, Age and AST. are positive, which makes sense to a casual basketball fan. Players that play more often and produce higher offensive statistics should be paid more. Additionally, players generally tend to sign larger contracts as their career progresses. However, G has a negative estimated coefficient, implying that on average, players are paid less money when they play in more games. This doesn't seem to be too logical at first. However, one could argue that a player that plays in more games would also generally play more minutes, and this argument would be backed by the 0.843 correlation coefficient between these two variables in our dataset. Perhaps one of these factors could have been removed early on, but given they are both highly statistically significant in our final model, it's probably good that they were both kept. The estimated coefficients specifically for MP and for G are 0.009451 and -0.0158993, respectively, implying that a player's predicted salary increases by approximately 0.09% for every extra minute played and decreases by about 1.58% for every extra game played. Therefore, the salary of a player who averages 18

or more minutes a game is predicted to benefit as the player plays more, while that of a player who averaged 17 minutes or less per game is expected to decrease as the player plays more.

At the 99% confidence level, two more variables are significant: USG. And Pos_catG. USG. has a positive estimated coefficient of 0.0244984, indicating that a player's expected salary increases 2.48% for every unit increase in USG. The Pos_catG indicator variable, conversely, has an estimated coefficient of -0.3685353. Since this model uses Pos_catBig as the baseline, this model projects that a guard's predicted salary will be more than 30% less than a center's given that the guard and center have identical factors for this model otherwise.

Two additional models are significant at the 90% confidence level: orb_adj and stl_adj, with a positive and negative estimated coefficient, respectively. The idea that a player who can accumulate more offensive rebounds is predicted to have a higher salary is completely logical. On the other hand, a player who garners more steals being predicted to have a lower salary may be a bit confusing for basketball fans. A possible explanation for this could be that players with more steals are more likely to be "defense-first" players, which in 2015-16, were not as highly valued as primarily offensive skilled players.

The estimated coefficient for the Pos_catWing indicator variable also becomes significant at the 90% confidence level, with an estimated value of -0.1741437. Like the situation for guards, our model proposes that the predicted salary for a Wing relative to a Center with the same model inputs otherwise is 15.98% lower, on average. The idea that talented bigs were much harder to come by than talented guards or wings during this time in the NBA would be widely agreed upon by fans. This could conceivably be why our model predicts centers with the same model-relevant statistics as a guard or wing would have higher salaries. Similarly, the talent pool was probably deepest among guards at this time, so there was less of an urge to pay guards as much money.

Also present in the model are Multiteam and TOV., though neither significant at the 90% level. The estimated coefficient for the Multiteam indicator variables is positive, suggesting players who played for various teams during the 2015-16 NBA season typically had higher salaries. The estimated coefficient for TOV. was negative, confirming the idea that players who turned the ball over more often were paid less, on average.

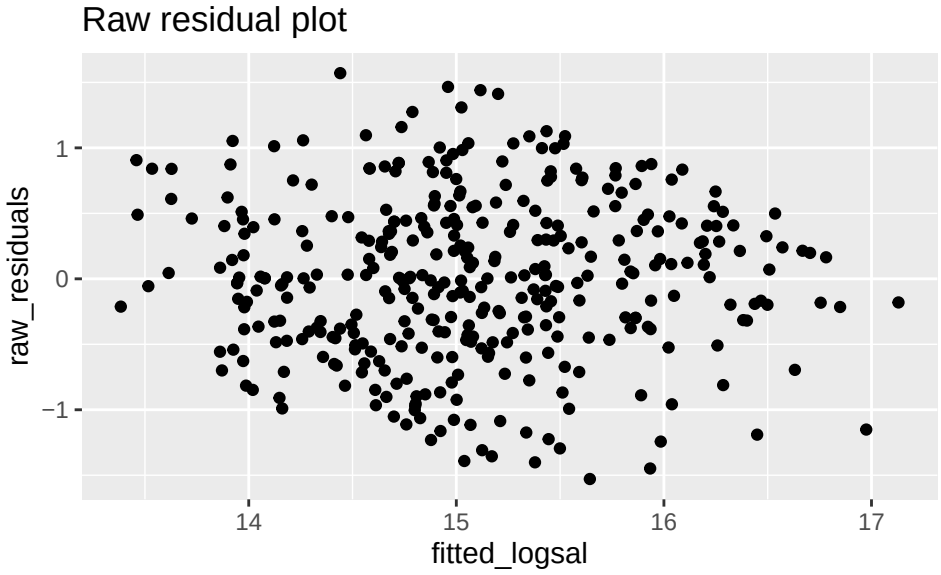
```
##
## Call:
## lm(formula = logsal ~ MP + Age + USG. + G + orb_adj + AST. +
##     stl_adj + Pos_cat + Multiteam + TOV., data = data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.52889 -0.41713  0.00419  0.42837  1.57068
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) 12.5466843  0.4046700  31.005  < 2e-16 ***
## MP           0.0009451  0.0000895  10.561  < 2e-16 ***
## Age          0.0673074  0.0077536   8.681  < 2e-16 ***
## USG.         0.0244984  0.0087791   2.791 0.005547 **
## G           -0.0158993  0.0031515  -5.045 7.26e-07 ***
## orb_adj      0.1679352  0.0747573   2.246 0.025294 *
## AST.         0.0220372  0.0060466   3.645 0.000308 ***
## stl_adj     -0.3190648  0.1307968  -2.439 0.015202 *
## Pos_catG    -0.3685353  0.1394006  -2.644 0.008565 **
## Pos_catWing -0.1741437  0.1021098  -1.705 0.088987 .
## Multiteam    0.1996041  0.1254847   1.591 0.112578
## TOV.        -0.0140422  0.0089122  -1.576 0.116007
## ---
```

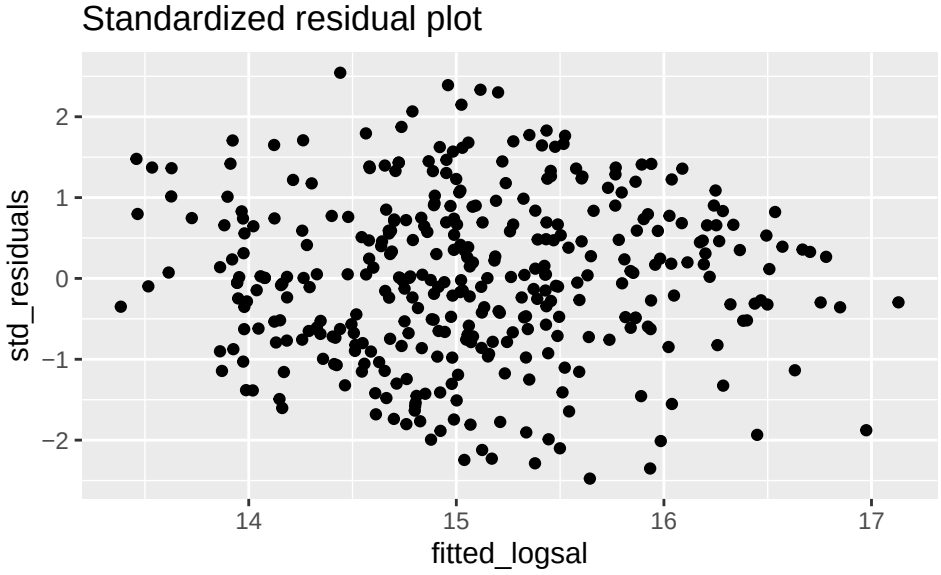
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6241 on 354 degrees of freedom
## Multiple R-squared:  0.5871, Adjusted R-squared:  0.5742
## F-statistic: 45.75 on 11 and 354 DF,  p-value: < 2.2e-16
```

Model Diagnostics

In all, our model reports an R^2 value of 0.5871 and Adjusted R^2 value of 0.5742. The 6-fold cross-validation procedure returns an average R^2 of 0.5665 on unseen test data, with the smallest root mean squared error and smallest mean absolute error of any of the five proposed models.

We will also look at the residual plot again, as well as the standardized residual plot:



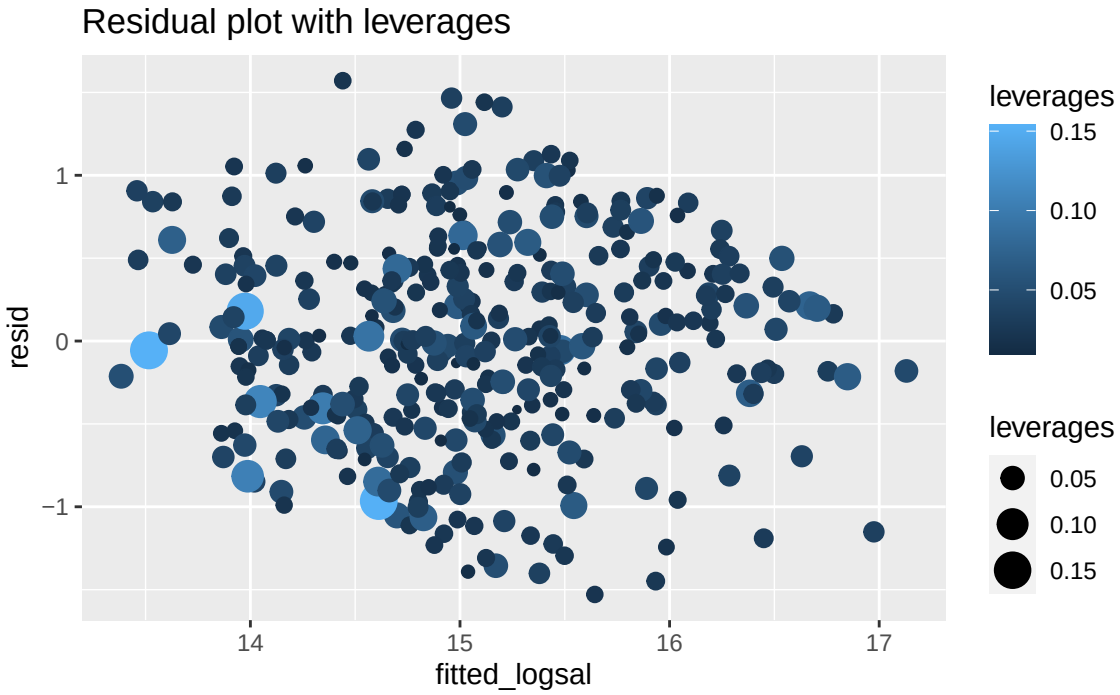


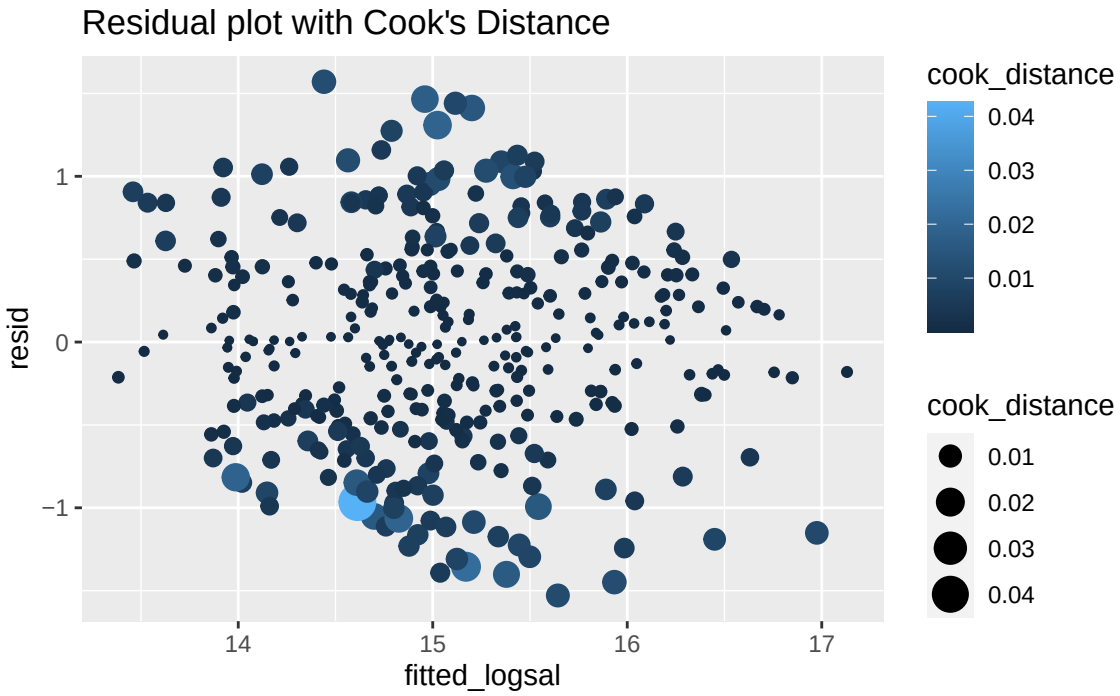
While these plots are not perfect, namely in that there appears to be a slightly downward trend and some possible heteroscedasticity, they are reasonable enough to suggest that none of the assumptions of the linear

model were flagrantly violated. The residuals are centered around zero, and somewhat resemble a "null plot." There does not appear to be any significant differences between the raw and standardized residuals that are worth discussing/exploring further.

Outliers, Cook, Leverage, etc.

We also looked at some of the highest leverage cases.





There's an outlying case with a large leverage and Cook's Distance. The residual does not appear to be

anything extraordinary, however. After digging into this data point, we see that it's Jarnell Stokes, a guy who appeared in 7 games and played 18 minutes. It does appear that he has some noisy statistics due to a small sample size from a lack of playing time. We'll use a t-test for a mean shift to formally test if this is an outlier.

```
## Outlier p-value:  0.04641536
```

This p-value is small, but not ridiculously small in the context of a dataset of 366 observations. We are unable to conclude that this point is an outlier and there is a reason to believe that the mean function for this point should be shifted.

There is another data point that is a bit of an outlier in terms of having the largest residual. It doesn't actually appear to be an outlier that suggests the mean function assumed by the model is incorrect, however. This player is Iman Shumpert; we'll discuss him a bit later in our conclusions section.

Contextual Applications

One thing that was of interest to us was using this model to see which players are overperforming and underperforming their salary, based on the model's expectations. We'll look at the largest and smallest residuals, raw and standardized:

```
## Most Underpaid: Rodney Hood, Expected Salary: 6220427, Actual Salary: 1348440
```

```
## Most Overpaid: Iman Shumpert:, Expected Salary: 1868802, Actual Salary: 8988765
```

Our model thought Rodney Hood was the league's most underpaid player, and Iman Shumpert was the league's most overpaid player (using both raw and standardized residuals, for both players). Hood's relatively high usage rate, offensive rebounding rate, and assists led to the model expecting him to have a fairly average salary, when he was actually paid towards the low end of the league. Conversely, Shumpert's classification as a guard, low usage rate and assists, and relatively high turnover rate led the model to think he should be paid fairly low, when in reality he was a well-paid player. We imagine that digging into which players were over/underpaid, and whether or not this had a correlation with team success, would be a fun and useful activity with a trustworthy model.

Conclusion

Appendix

Figure A1: Transformation of VORP

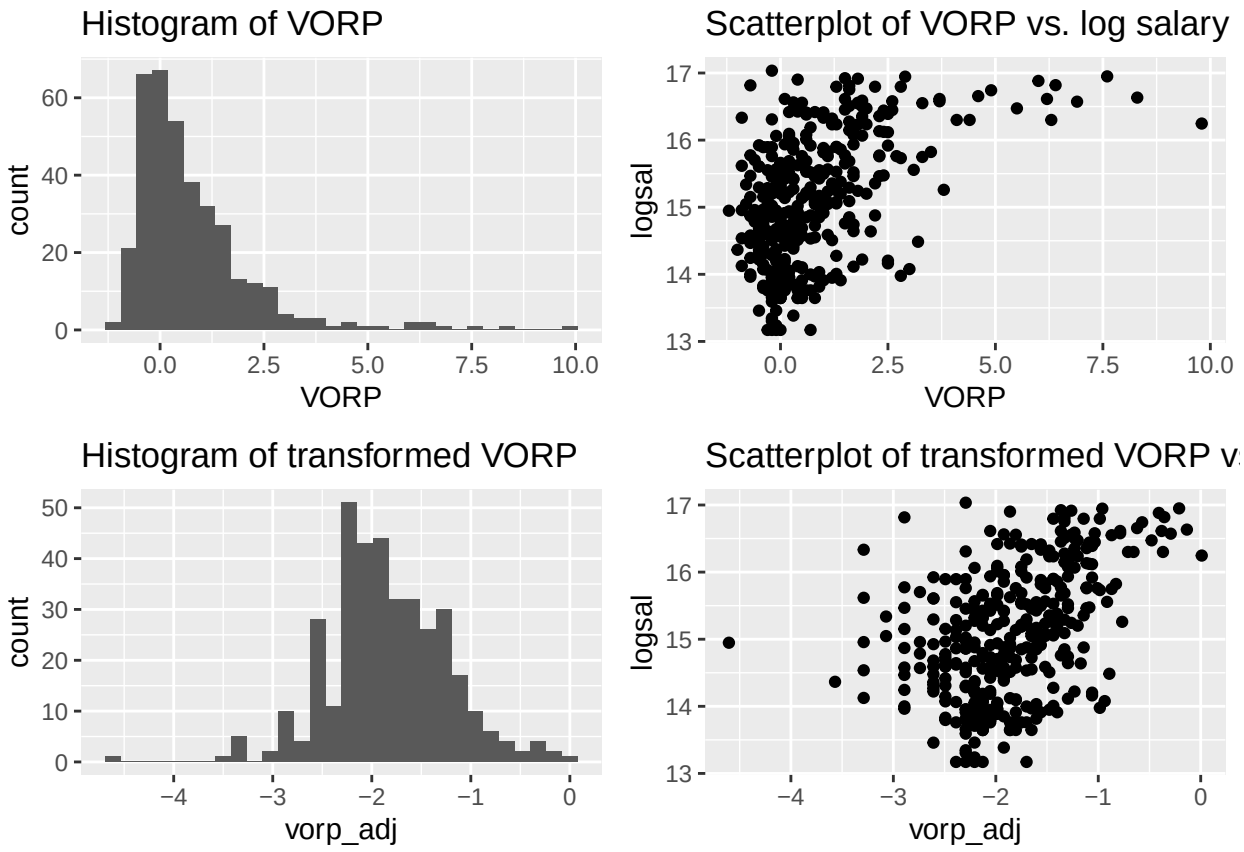
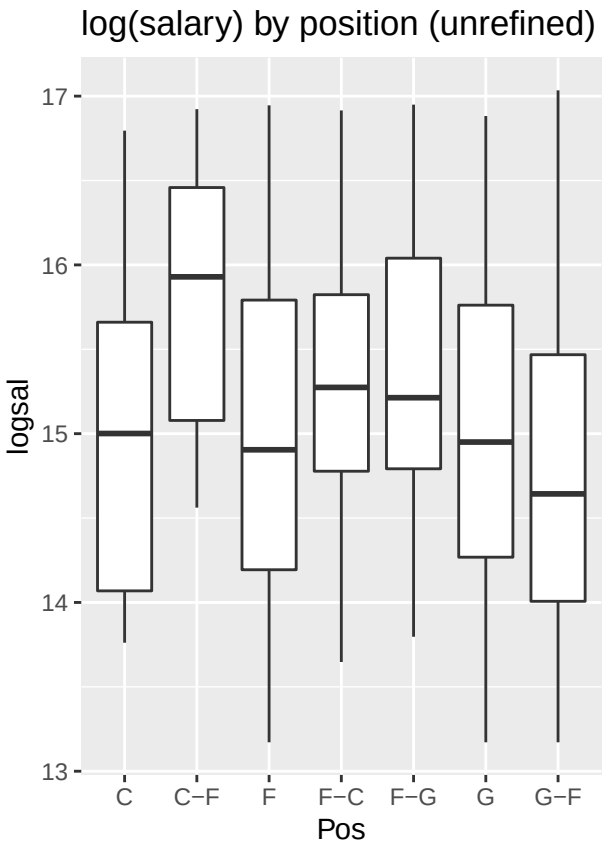
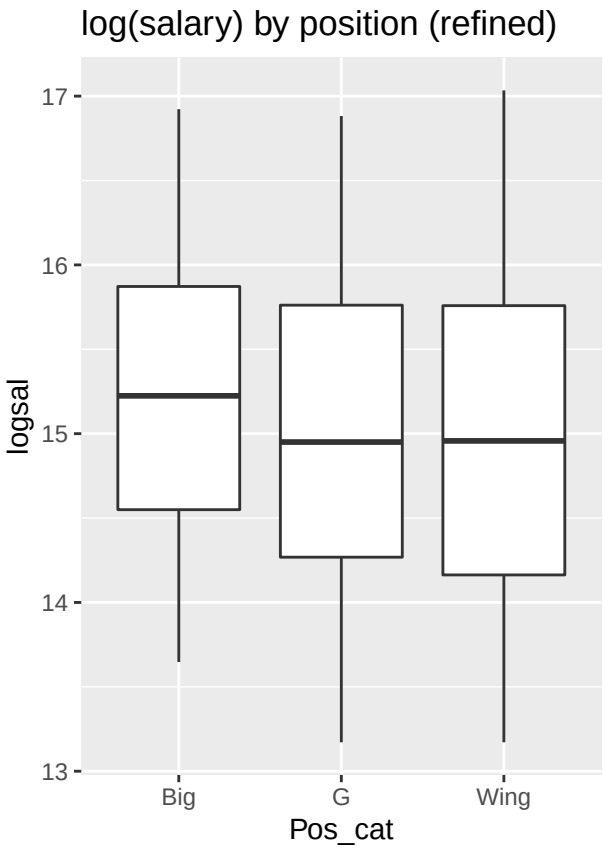


Figure A2: Transformation of Pos


log(salary) by position (unrefined)
logsal
17
16
15
14
13
C
C−F
F
F−C
F−G
G
G−F
Pos


log(salary) by position (refined)
logsal
17
16
15
14
13
Big
G
Wing
Pos_cat