

# Analyze Results

Basketball Salaries Team

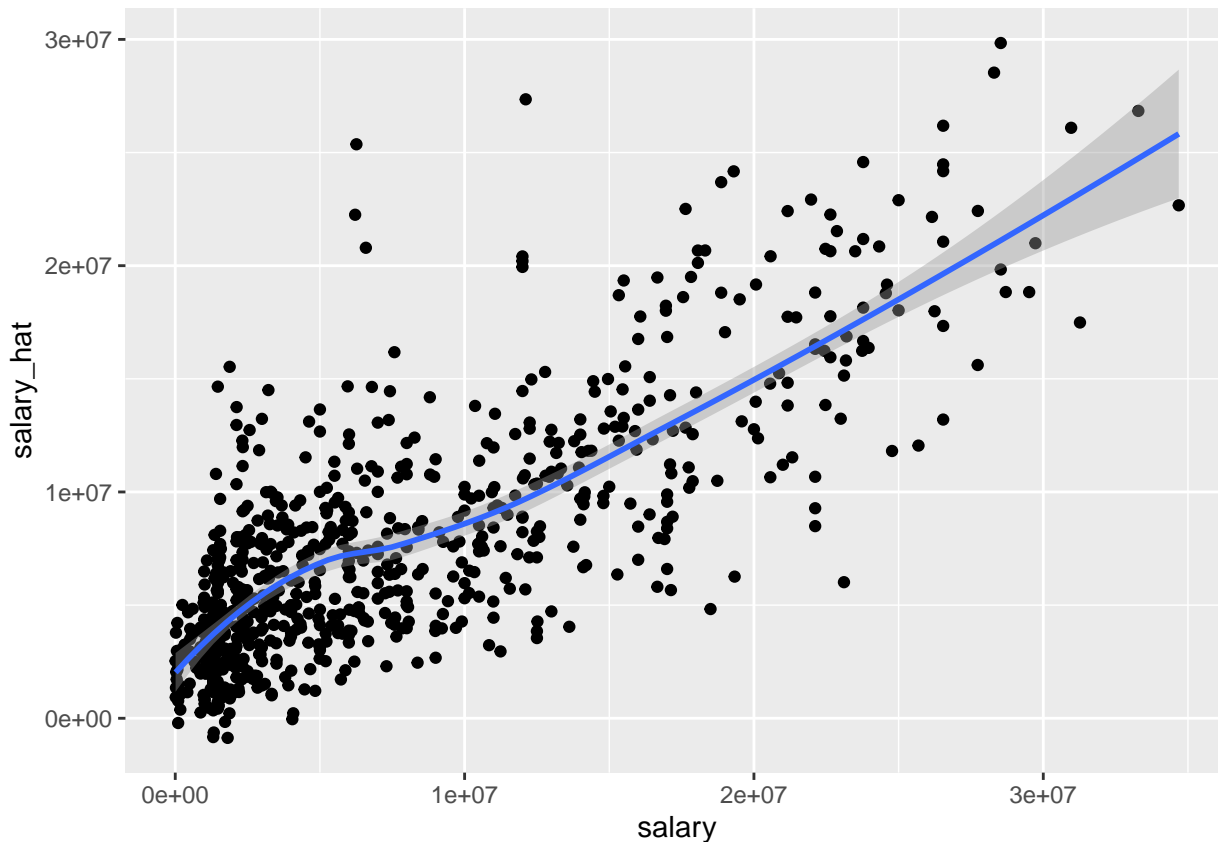
## Load Results from Optimal Elasticnet Model

```
df <- read.csv('../data/predictions/elasticnet.csv')
df[df$name=='lebron james',]
```

```
##           name year  salary salary_hat
## 440 lebron james 2016 30963450  26093545
## 441 lebron james 2017 33285709  26839670
```

## Plot Salary vs Predicted Salary

```
library(ggplot2)
ggplot(df, aes(x=salary, y=salary_hat)) +
  geom_point() +
  geom_smooth()
```

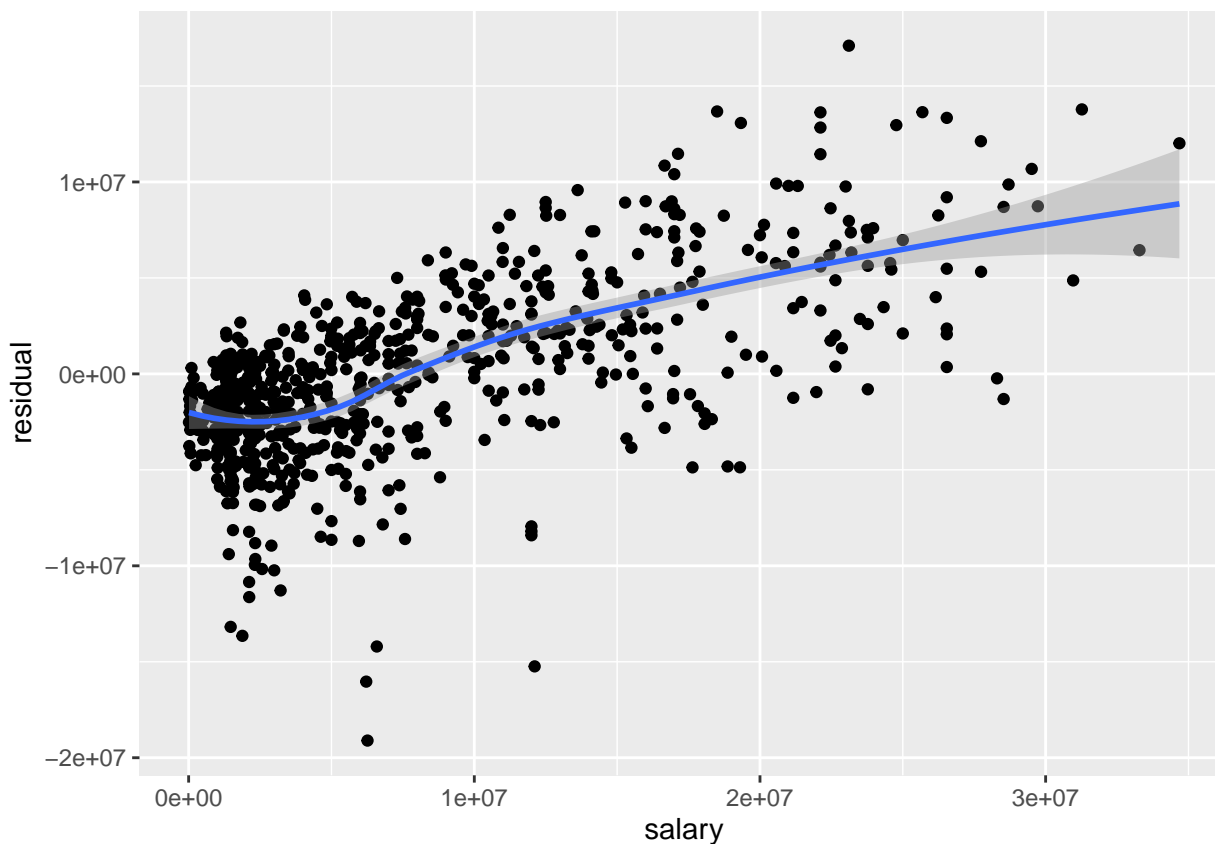


```
ggsave("../figures/enet_salary_vs_prediction_scatter.png", width=10, height=7)
```

## Plot Salary vs Residual Prediction

Appears to be a positive linear trend between the salary and residuals.  
Thus, the optimal elasticnet model tends to underestimate players with higher salaries.

```
library(ggplot2)
residual <- df$salary - df$salary_hat
ggplot(df, aes(x=salary,y=residual)) +
  geom_point() +
  geom_smooth()
```



```
ggsave("../figures/enet_salary_vs_residual_scatter.png", width=10, height=7)
```

## Underrated Players in 2016

An underrated player is any player who should be making 50% more

$$\hat{\text{salary}} - \text{salary} > \frac{1}{2}\text{salary}$$

Players who our model classifies as underrated tend to get higher salaries the next year

```
# get underrated players in 2016
df_underrated_2016 <- df[
  ((df$salary_hat-df$salary) > (df$salary/2)) &
  df$year==2016,]
# link with 2017 stats
df_underrated_2017 <- df[(df$name%in%df_underrated_2016$name)&(df$year==2017),]
df_underrated <- merge(df_underrated_2016,df_underrated_2017,all.x=F,all.y=F,by='name')
names(df_underrated) <- c('name','2016','salary_2016','salary_hat_2016','2017','salary_2017','salary_hat_2017')
df_underrated <- df_underrated[,c('name','salary_2016','salary_hat_2016','salary_2017','salary_hat_2017')]
df_underrated
```

##	name	salary_2016	salary_hat_2016	salary_2017
## 1	aaron gordon	4351320	8286104.8	5504420
## 2	anderson varejao	1984005	4735387.9	1913345
## 3	andre roberston	2183072	5797383.2	9259259
## 4	andrew wiggins	6006600	12541957.7	7574322
## 5	bobby portis	1453680	3117239.5	1516320
## 6	bojan bogdanovic	3730653	6499120.2	10500000
## 7	cj mccollum	3219579	14503449.7	23962573
## 8	cj miles	4583450	7388745.7	7936509
## 9	clint capela	1296240	5114481.5	2334520
## 10	darren collison	5229454	9055869.1	10000000

## 11	david west	1551659	9693036.1	2328652
## 12	delon wright	1577280	2366257.0	1645200
## 13	dennis schroder	2708582	6860532.1	15500000
## 14	devin booker	2223600	5362632.1	2319360
## 15	dion waiters	2898000	6151961.4	11000000
## 16	doug mcdermott	2483040	5660426.2	3294994
## 17	elfrid payton	2613600	7793728.1	3332340
## 18	emmanuel mudiay	3241800	6205590.9	3381480
## 19	frank kaminsky	2730000	5289751.4	2847600
## 20	gary harris	1655880	6225110.1	2550055
## 21	george hill	8000000	12163220.1	20000000
## 22	giannis antetokounmpo	2995421	13231011.4	22471910
## 23	gorgui dieng	2348783	7738226.0	14112360
## 24	isaiah canaan	1015696	4916526.2	2000000
## 25	isaiah thomas	6587132	20791520.6	6261395
## 26	ish smith	6000000	9098922.9	6000000
## 27	jabari parker	5374320	8449932.1	6782392
## 28	jae crowder	6286408	11028598.6	6796117
## 29	jahlil okafor	4788840	7670507.3	4995120
## 30	jarell eddie	175000	380587.1	17224
## 31	jarrett jack	1551659	7700472.4	2328652
## 32	jason terry	1551659	4657437.1	2328652
## 33	javale mcgee	1403611	4245730.8	2116955
## 34	jeff teague	8800000	14185938.8	19000000
## 35	jeff withey	1015696	2839837.5	1577320
## 36	jerami grant	980431	5104947.7	1524305
## 37	jj barea	4096950	8211377.8	3903900
## 38	jj redick	7377500	13180799.4	23000000
## 39	joffrey lauvergne	1709720	4661511.9	1524305
## 40	jonathon simmons	874636	2956668.4	6300000
## 41	josh huestis	1191480	3969558.7	1471382
## 42	josh richardson	874636	2145132.8	1471382
## 43	julius randle	3267120	8574670.6	4149242
## 44	justin holiday	1015696	2963229.5	4615385
## 45	justise winslow	2593440	5179143.0	2705040
## 46	jusuf nurkic	1921320	3065890.6	2947305
## 47	karl anthony towns	5960160	14666696.1	6216840
## 48	kelly olynky	3094014	6291168.4	10607169
## 49	kemba walker	12000000	20410535.3	12000000
## 50	kentavious caldwell pope	3678319	9404962.4	17745894
## 51	kristaps porzingis	4317720	9630442.9	4503600
## 52	kyle anderson	1192080	3360877.2	2151704
## 53	kyle korver	5239437	10183482.6	7000000
## 54	kyle lowry	12000000	20204144.3	28703704
## 55	lou williams	7000000	10902031.6	7000000
## 56	marcus morris	4625000	13111574.3	5000000
## 57	marcus smart	3578880	5644722.3	4538020
## 58	marreese speights	1403611	3666717.4	2116955
## 59	mason plumlee	2328530	11146506.1	14041096
## 60	michael beasley	1403611	3405414.8	2116955
## 61	michael carter williams	3183526	6173725.7	2700000
## 62	myles turner	2463840	5056516.9	2569920
## 63	nerlens noel	4384490	7195523.2	4187599
## 64	nikola jokic	1358500	8103074.3	1471382
## 65	norman powell	874636	4010833.8	1471382
## 66	omri casspi	3000000	7961592.4	2106470
## 67	paul pierce	3500000	7156731.7	1096080
## 68	pj tucker	5300000	8229036.4	7590035
## 69	rakeem christmas	1052342	2307327.7	172552
## 70	raul neto	937800	3826166.3	1471382
## 71	raymond felton	1551659	8284117.5	2328652
## 72	richard jefferson	2500000	5181166.1	2500000
## 73	robert covington	1015696	5909399.1	16698103
## 74	rodney hood	1406520	10797855.3	2386864

## 75	rondae hollis jefferson	1395600	3762800.2	1471382
## 76	rudy gobert	2121288	13751283.4	21974719
## 77	salah mejri	874636	3144578.7	1471382
## 78	sean kilpatrick	980431	2674652.0	1524305
## 79	shabazz muhammad	3046299	5431883.5	1577230
## 80	shelvin mack	2433334	5141093.6	6000000
## 81	spencer dinwiddie	726672	2228495.3	1524305
## 82	stephen curry	12112359	27352365.8	34682550
## 83	steven adams	3140517	9991738.3	22471910
## 84	thabo sefolosha	3850000	8346980.1	5250000
## 85	tj warren	2128920	3544866.5	3152931
## 86	tony snell	2368327	4040666.9	9821429
## 87	trey lyles	2340600	4433565.8	2441400
## 88	victor oladipo	6552960	10501995.5	21000000
## 89	will barton	3533333	9762814.1	3533333
## 90	willie cauley stein	3551160	6444626.7	3704160
## 91	zach lavine	2240880	7998179.6	3202217
## 92	zaza pachulia	2898000	11845725.8	3477600
##	salary_hat_2017			
## 1	10716913.8			
## 2	6498688.3			
## 3	7794637.6			
## 4	16177279.2			
## 5	3768595.2			
## 6	9833913.3			
## 7	16369999.9			
## 8	8358124.9			
## 9	11980005.9			
## 10	9178202.9			
## 11	7339153.9			
## 12	2316848.3			
## 13	13273554.7			
## 14	12270237.2			
## 15	9295510.5			
## 16	7698529.4			
## 17	9961869.6			
## 18	5591950.2			
## 19	8737618.0			
## 20	8312533.9			
## 21	12772197.7			
## 22	20743921.8			
## 23	9455761.2			
## 24	1084718.8			
## 25	25366136.3			
## 26	8098892.2			
## 27	11137029.8			
## 28	14641353.1			
## 29	7476043.5			
## 30	929282.4			
## 31	2286364.1			
## 32	5654844.9			
## 33	7816633.1			
## 34	17062262.2			
## 35	2426601.6			
## 36	4262138.2			
## 37	8552614.0			
## 38	13236926.3			
## 39	3461134.7			
## 40	5580168.4			
## 41	4321017.6			
## 42	6007926.0			
## 43	9402205.1			
## 44	4629632.2			
## 45	4692807.8			

```
## 46      7703827.9
## 47     22250990.1
## 48      8031695.6
## 49     19945218.8
## 50     11085004.6
## 51     11529715.6
## 52      2583328.5
## 53     10005116.9
## 54     18835023.8
## 55     13064185.7
## 56     12671675.4
## 57      8428998.6
## 58      5709406.8
## 59     11760572.6
## 60      5214431.1
## 61      3785920.9
## 62     12737848.8
## 63      5498705.6
## 64     14653690.1
## 65      6108513.1
## 66      4054661.7
## 67      6973136.0
## 68      6380338.6
## 69      2543977.9
## 70      426385.8
## 71      5224175.0
## 72      7727971.5
## 73      7968749.0
## 74      9208872.2
## 75      7063077.4
## 76     22923156.7
## 77      3345920.2
## 78      6978799.8
## 79      5013230.7
## 80      3683850.5
## 81      3483574.5
## 82     22668263.3
## 83     13844144.6
## 84      9186163.8
## 85      7458134.3
## 86      7810897.8
## 87      3516974.4
## 88     11202922.6
## 89      8520207.9
## 90      8870964.3
## 91      8957523.8
## 92      9598059.8
```

```
# calculate percent increase factor
sal_factor_2016 = .25
p_increase <- 100*mean(
  (df_underrated$salary_2017-df_underrated$salary_2016) > sal_factor_2016*df_underrated$salary_2016)
sprintf(
  'Percent of underrated players in 2016 whose salary increased by more than 25%% for 2017: %.1f%%',p_increase)

## [1] "Percent of underrated players in 2016 whose salary increased by more than 25% for 2017: 62.0%"
```

## OVERRATED PLAYERS IN 2016

An overrated player is any player who should be making 50% more

$$salary - \hat{salary} > \frac{1}{2} salary$$

Hard for players who have high salaries to get a salary cut despite lacking stats

```
df_overrated_216 <- df[
  ((df$salary-df$salary_hat) > (df$salary/2)) &
  df$year==2016,]
# link with 2017 stats
df_overrated_2017 <- df[(df$name%in%df_overrated_216$name)&(df$year==2017),]
df_overrated <- merge(df_overrated_216,df_overrated_2017,all.x=F,all.y=F,by='name')
names(df_overrated) <- c('name','2016','salary_2016','salary_hat_2016','2017','salary_2017','salary_hat_2017')
df_overrated <- df_overrated[,c('name','salary_2016','salary_hat_2016','salary_2017','salary_hat_2017')]
df_overrated
```

	name	salary_2016	salary_hat_2016	salary_2017
## 1	allen crabbe	18500000	4826359.9	19332500
## 2	austin rivers	11000000	4444805.5	11825000
## 3	bradley beal	22116750	8490691.4	23775506
## 4	chandler parsons	22116750	10671332.6	23112004
## 5	cole aldrich	7643979	3609821.7	7300000
## 6	demarre carroll	14200000	6769843.2	14800000
## 7	dwight powell	8375000	2455833.5	9003125
## 8	harrison barnes	22116750	9281186.3	23112004
## 9	iman shumpert	9700000	3994809.8	10337079
## 10	joakim noah	17000000	8393785.7	17765000
## 11	john henson	12517606	4276013.1	11422536
## 12	jon leuer	10991957	5157916.2	10497319
## 13	kevon looney	1182840	451297.3	1471382
## 14	kj mcdaniels	3333333	1013608.2	100000
## 15	kyle singler	4837500	1211564.5	4666500
## 16	mario hezonja	3909840	1454530.1	4078320
## 17	maurice harkless	8988764	3857698.1	10162922
## 18	meyers leonard	9213484	3971522.2	9904495
## 19	michael kidd gilchrist	13000000	4720726.3	13000000
## 20	mike conley	26540100	13199859.9	28530608
## 21	miles plumlee	12500000	3849374.0	12500000
## 22	nemanja bjelica	3800000	1814264.4	3949999
## 23	pat connaughton	874636	255027.1	1471382
## 24	rashad vaughn	1811040	-866999.7	1889040
## 25	sam dekker	1720560	-160575.3	1794600
## 26	solomon hill	11241218	2953446.4	12236535
## 27	tarik black	6191000	2502750.3	3290000
## 28	timofey mozgov	16000000	7002922.3	15280000
## 29	trevor booker	9250000	4605976.8	9125000
## 30	troy daniels	3332940	1080160.7	3408520
## 31	tyler zeller	8000000	3979985.2	1709538
##	salary_hat_2017			
## 1		6259010.6		
## 2		7250777.4		
## 3		16673157.2		
## 4		6012393.7		
## 5		2297919.6		
## 6		9827352.9		
## 7		4086057.0		
## 8		15143129.4		
## 9		6458411.9		
## 10		10186031.6		
## 11		6202557.1		
## 12		8525008.0		
## 13		465991.2		
## 14		768466.9		
## 15		2169666.7		
## 16		219604.2		
## 17		6521070.7		
## 18		4274749.9		
## 19		10893707.4		
## 20		19832043.2		

```
## 21      3544135.9
## 22      3426925.7
## 23       601458.5
## 24       875860.4
## 25      4256985.8
## 26      7109339.3
## 27      3436066.5
## 28      6354677.0
## 29      8216626.0
## 30      4127478.8
## 31      3084342.7
```

```
# calculate percent decrease factor
```

```
sal_factor_2016 = .0
```

```
p_decrease<- 100*mean(
```

```
  (df_overrated$salary_2016-df_overrated$salary_2017) > sal_factor_2016*df_overrated$salary_2016)
```

```
sprintf(
  'Percent of overrated players in 2016 whose salary decreased by more than 25%% for 2017: %.1f%%',p_decrease)
```

```
## [1] "Percent of overrated players in 2016 whose salary decreased by more than 25% for 2017: 29.0%"
```