Trabajo Especializacion

Franco Betteo

01/09/2019

Introducción

No hay una manera única e inequívoca de comparar rendimientos de jugadores en los distintos deportes y eso da lugar a discusiones sin fin. Más díficil aún en los deportes en equipo donde hay roles diferentes y contribuciones de distinta índole. A raíz de esto se han ido desarrollando métricas que intentan resumir el aporte al equipo de manera integral para hacer comparables los jugadores. En básquetbol una de las medidas más conocidas de este tipo es el "plus-minus" (originalmente implementado en el hockey sobre hielo), que calcula la diferencia de puntos de un equipo mientras cada jugador estuvo en cancha. Es decir que valores positivos (negativos) revelan que durante un partido el equipo hizo más (menos) puntos de los que recibió mientras el jugador estuvo en cancha. Es una métrica sencilla de calcular y resume el aspecto más importante de un partido de manera general para cada jugador. Es fácil de interpretar pero no está exenta de problemas. La idea de este trabajo es aplicar un método más robusto, "Plus-Minus ajustado", basado en los aportes de Justin Jacobs[1] y Joseph Sill[2], donde el principal agregado es controlar por los otros jugadores en cancha. El objetivo es actualizar la métrica para la temporada 2017-2018 y generar un ranking de jugadores. Posteriormente comparar el ranking contra otros generados por algún medio conocido para ver similitudes y diferencias. Por último generar un ranking de equipos, basado en los rankings individuales de jugadores, y comparar contra los resultados de las rondas definitorias del certamen cuyos datos no son utilizados para calcular la métrica.

Datos

Para poder calcular la métrica "plus minus ajustado" necesitamos tener para cada momento del partido los jugadores que hay en cancha y el resultado ya que el objetivo es ver la performance del equipo en presencia y ausencia de cada jugador de la liga. A tales fines se decidió utilizar la información provista por la API de MySportsFeed[3]. En ella podemos encontrar datos a un nivel suficientemente granular. En particular, para cada partido tenemos información jugada a jugada marcada por ciertas situaciones particulares, entre ellas tiros al aro y si se conviertieron puntos o no, rebotes, faltas y sustituciones. Con la primera y la última de estas características podemos recolectar los datos necesarios para generar nuestro dataset. Al estar todas las sustituciones y tener la alineación inicial de cada equipo podemos obtener todos los segmentos del partido donde hubo distintas combinaciones de jugadores en cancha (tanto compañeros como rivales) para cada jugador. A su vez, al tener los puntos anotados podemos obtener el diferencial de puntos para cada uno de estos segmentos.

A los fines de utilizar toda esta información para entrenar el modelo de "plus minus ajustado" necesitamos armar un dataset con el formato siguiente:

- Cada observación es un segmento de un partido donde se mantuvo constante la alineación de ambos equipos.
- Las variables independientes son cada uno de los jugadores de la liga, con valor de 1 si estaban en cancha en ese segmento para el equipo local y -1 si estaban en cancha siendo del equipo visitante.
- La variable dependiente es el diferencial de puntos del equipo local. Valores positivos (negativos) es que anotó más puntos el equipo local (visitante).

Esta tabla esta conformada con todos los equipos de la liga y para cada partido de la temporada, de manera que para apendizar la información de cada encuentro hay que tener las variables independientes de cada equipo, es decir, todos sus jugadores. Dado que cada partido solo involucra a dos equipos, la gran mayoría de

[1] https://squared2020.com/2017/09/18/deep-dive-on-regularized-adjusted-plus-minus-i-introductory-example/
[2] http://www.sloansportsconference.com/wp-content/uploads/2015/09/joeSillSloanSportsPaperWithLogo.pdf
[3] https://www.mysportsfeeds.com/data-feeds/api-docs

las columnas tendrán valor de 0 en cada observación, siendo el dataset una matriz dispersa (sparse matrix). Se toman valores de 1 y -1 según la localía para que el signo quede acorde a la medición del diferencial de puntos y por ende el signo de los coeficientes sea siempre positivo para aportes beneficiosos a un equipo y negativos para aportes perjudiciales.

Tanto para la consulta de la API como para el manejo y procesamiento de los datos se utilizó el lenguaje R [4]

INCLUIR HEAD DE LA TABLA O MANDAR A APENDICE

Modelo

La metodología para calcular la métrica "plus minus ajustado" implica correr una regresión Ridge (regularización con *l2*) donde la variable dependiente es el diferencial de puntos por segmento y las variables independientes son los jugadores en cancha. La idea de fondo es calcular el aporte de cada jugador al diferencial, controlando por sus compañeros y por los advsersarios. Se intenta eliminar el factor que sobreestima el aporte de algún jugador solo por el hecho de compartir tiempo en cancha con compañeros de primer nivel, que son los que generan realmente los diferenciales positivos. De la misma manera se intenta no sobreestimar a los jugadores que anotan muchos puntos contra equipos de baja performance o que solo juegan en los minutos llamados "basura" que corresponden a los minutos finales de un partido cuando ya está todo definido y suelen haber jugadores de menor nivel. Dado que es una regresión lineal los coeficientes pueden interpretarse como un proxy del aporte neto de cada jugador, ya descontados los aportes del resto. Dada la codificación de variables postiva para el local y negativa para visitantes, todo signo positivo de un coeficiente es aporte real en puntos y negativo es tendencia a recibir más puntos que los que se convierten con el jugador en cancha.
Los equipos suelen tener una plantilla de alrededor de 10 jugadores activos y los que más minutos disputan suelen ser menos aún por lo tanto es de esperar que haya una gran correlación entre los jugadores de cada equipo. Este es el principal motivo por el que se decide ir por una regresión regularizada y no una regresión lineal multivariada clásica.
Ridge lo que hace es reducir la varianza de los coeficientes incluyendo una penalización *l2* que implica reducir mínimos cuadrados sumado a la diferencia al cuadrado de los coeficientes respecto de 0. Esto último ponderado por un parámetro λ a definir. A mayor λ la penalización es mayor y los coeficientes tienden a valores cercanos a 0. El procedimiento reduce la varianza de los coeficientes a costa de introducir un sesgo en su estimación, pero que de tener éxito, el tradeoff es tal que las predicciones son más certeras a pesar de no ser insesgado.

Formalización

Formalmente el modelo especificado es:

$$Y = \sum_{1}^{n} \beta_i X_i + \epsilon$$

Donde Y es el diferencial de puntos visto desde el equipo local y las X son las variables de presencia/ausencia de cada jugador del partido. Siendo estrictos, en la matriz estarán todos los jugadores de la liga por lo que las X incluyen a muchísimos jugadores que no son parte del partido pero obviamente tendrán valor 0 y no tendrán injerencia en la suma. Los β son los coeficientes de la regresión.

Dado nuestro modelo, lo siguiente es preguntarse cómo estimar los coeficientes con los datos que tenemos a disposición. Mínimos cuadrados ordinarios (MCO) es posiblemente la primera opción dado que sus coeficientes tienen propiedades interesantes: los estimadores son insesgados y si se cumplen ciertas condiciones sobre los errores del modelo también son los estimadores de mínima varianza dentro de los insesgados.

[4] R Core Team (2013). R: A language and environment for statisticalcomputing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/

Los estimadores de MCO se obtienen mediante

$$\hat{\beta} = argmin\ S(\beta)$$

donde

$$S(\beta) = \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2$$

A pesar de mantener la propiedad de insesgadez, estos estimadores son bastante sensibles a la alta correlación entre variables ya que al existir en planos muy próximos existen mayor cantidad de combinaciones lineales entre ambas variables que dan resultados similares en cuanto al ajuste del modelo. Esto se traduce en que los estimadores de MCO puedan variar mucho entre distintas muestras de la misma población, es decir que son estimadores con varianza elevada. Aunque en promedio los estimadores se centren en los verdaderos valores de β, lo van a hacer de manera muy errática y los valores que encontremos en nuestra muestra no van a ser confiables y por lo tanto nuestras predicciones tampoco.

En este momento es donde hay que tener en mente el tradeoff entre sesgo y varianza de un modelo. Este concepto nos dice que el error cuadrático medio (MSE por sus siglas en inglés) de un set de testeo puede descomponerse entre sesgo elevado al cuadrado más varianza - del modelo estimado y aplicada a la nueva observación - más la varianza del error irreducible del proceso generador de datos.
El último elemento está fijo pero los primeros dos varían según el modelo utilizado y estimado. En general modelos más flexibles tienden a tener menor sesgo ya que pueden ajustarse mejor a las no linealidades de los datos pero a su vez suelen tener más varianza ya que cambios en los datos tienen impacto sobre cómo ajustan entre muestras distintas. Dijimos que MCO sufre de alta varianza en sus estimadores ante presencia de variables correlacionadas y como mencionamos, el dataset de este trabajo presenta tal característica ya que los jugadores de un equipo son pocos y suelen compartir minutos en cancha de manera reiterada. La propuesta de utilizar una regresión Ridge apunta a tratar de sobrepasar la problemática de la alta varianza. El mecanismo es que Ridge agrega una penalización a la minimización del desvío cuadrático respecto a cero y eso genera dos consecuencias:

- Los estimadores dejan de ser insesgados ya que se los restringe al penalizar su alejamiento de 0.
- Los estimadores ven su varianza reducida ya que se limita el espacio de búsqueda.

Formalmente la minimización es la siguiente:

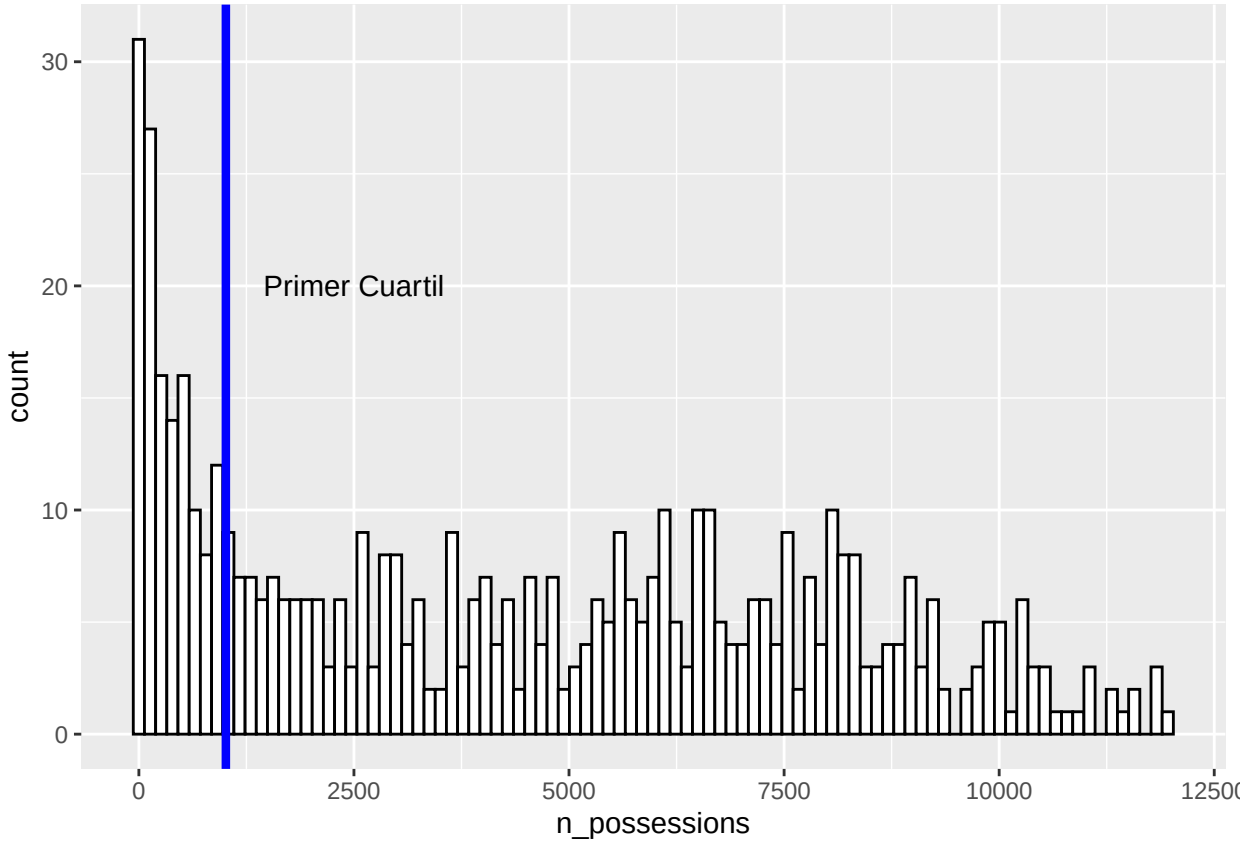
$$S(\beta) = \sum_{i=1}^{n} (y_i - \sum_{j=1}^{p} X_{ij}\beta_j)^2 + \lambda \sum_{j=1}^{p} \beta_j^2$$

Donde λ es un hiperparámetro que regula el peso de la penalización en la función de costo. Mayores (menores) valores de λ hacen más (menos) costosos los desvíos de los estimadores respecto de cero. En los extremos, si $\lambda = 0$, los estimadores son los mismos que MCO y si $\lambda = +\infty$ todos los estimadores son 0 y queda solo el intercepto. El objetivo de incluir la penalización es agregar un poco de sesgo a cambio de reducir en mayor medida la componente de varianza del MSE mediante la restricción que se aplica sobre los coeficientes estimados. El éxito de este enfoque depende en gran medida del λ elegido. Para seleccionar el valor de λ la práctica habitual es hiperparametrizar el modelo medianto crossvalidation.

Aplicación

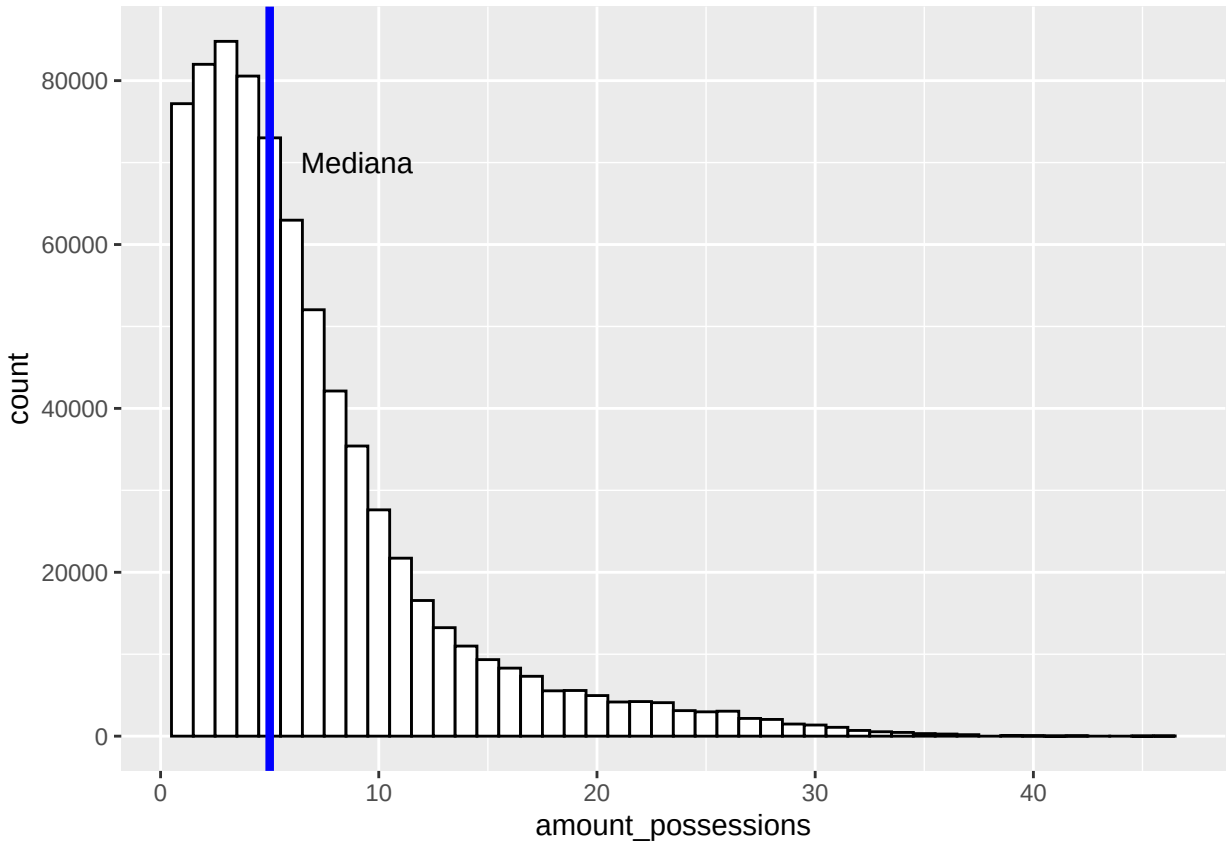
Mencionamos a continuación algunos aspectos metodológicos a tener en cuenta:

- Se eliminó el primer cuartil de jugadores medido en cantidad de posesiones en toda la temporada para reducir la dimensionalidad del problema. Son jugadores con pocos minutos en cancha y perjudican más de lo que aportan.



- Se eliminaron las observaciones con menos de 5 posesiones. Este número corresponde a la mediana de la cantidad de posesiones por observación. Se considera que observaciones con menos de esa cantidad de posesiones no son representativas y aumentan el ruido dentro del modelo.

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   3.000   5.000   6.831   9.000  46.000
```



- Se normalizaron los datos de diferencial de puntos cada 100 posesiones para hacer comparables los segmentos de tiempo.

  Una vez preprocesados todos los datos y habiendo definido el modelo, procedemos a la etapa de correr el modelo. La librería que vamos a utilizar es *glmnet*[5].
  Dijimos que uno de los hiperparámetros de la regresión Ridge es λ que define el peso de la penalización. Lo primero que hacemos es definir qué valor va a tomar para nuestro problema. La solución propuesta es el calcular el λ óptimo mediante Cross Validation con 10 folds. La librería provée la función *cv.glmnet* para tal fin, donde lo que hace es dividir el set de datos en 10 subsets y entrenar un modelo con 9 de los 10 subsets y validarlo con el décimo. Así para cada combinación de subsets. Esto lo realiza reiteradas veces para distintos valores de λ. Finalmente calcula el error cuadrático medio para cada λ en los sets de validación. Entre todos esos resultados buscamos el λ cuyo error promedio en validación sea el menor y será el que utilicemos para nuestro modelo con todos los datos.

Al aplicar este procedimiento obtenemos el óptimo para nuestro set de datos: $\lambda = 1.369778$

Tomando este valor de λ, lo siguiente es correr el modelo Ridge con todo el training set imputando ese valor para el hiperparámetro.

A continuación mostramos los primeros diez jugadores según el coeficiente obtenido (de mayor a menor), donde un coeficiente más elevado está asociado a mayor diferencial de puntos a favor mientras se está en cancha, controlando por el resto de los jugadores presentes (compañeros como oponentes).

[5]Jerome Friedman, Trevor Hastie, Robert Tibshirani (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. Journal of Statistical Software, 33(1), 1-22. URL http://www.jstatsoft.org/v33/i01/.

playerid	player.firstName	player.lastName	team.abbreviation	coef
9418	Joel	Embiid	PHI	2.173248
9420	Robert	Covington	PHI	2.159759
9402	Victor	Oladipo	IND	2.108312
9265	Chris	Paul	HOU	2.019688
9352	Eric	Gordon	HOU	1.932603
9218	Stephen	Curry	GSW	1.791738
9325	Giannis	Antetokounmpo	MIL	1.784740
9152	Jimmy	Butler	MIN	1.776535
9524	Otto	Porter Jr.	WAS	1.705704
9224	Shaun	Livingston	GSW	1.651169

Puede discutirse el orden y si falta algún jugador importante de la liga pero no es un primer ranking totalmente descabellado. Son jugadores de primer nivel y varios de ellos son "indiscutidos". Es destacable que en 10 jugadores haya 3 duplas que comparten equipo. Es posible que esté relacionado a cierta correlación en la presencia de estos jugadores en cancha y a su vez se de en equipos que han tenido muchas victorias en la temporada. Covington, Gordon y Livingston son de alguna manera una sorpresa en este ranking.

Viendo el ranking surge una nueva pregunta. ¿Qué tan precisos son esos coeficientes? ¿ Qué tan estable es ese ranking?
La manera de cuantificar esto en una regresión lineal es calcular los desvíos estándar de los coeficientes estimados. Hay un punto muy importante a mencionar respecto a esto. La regresión Ridge es lineal en parámetros pero como dijimos, incluye una penalización y eso lleva a que los coeficientes sean sesgados. Dicho esto, los desvíos estándar que calculemos serán sobre parámetros estimados que no sabemos en principio qué tan lejos están en promedio de los verdaderos parámetros si el modelo especificado fuera el correcto. Es posible un escenario donde gran parte del MSE de nuestra regresión se de por el sesgo y la varianza no contribuya casi, dando la impresión de gran precisión en la estimación ignorando el sesgo que introdujimos.(Goeman et al. 2018)[6]

Teniendo en cuenta esta limitación a la hora de interpretar los resultados pasamos al cálculo de los desvíos estándar. El paquete *glmnet* no tiene una función que devuelva los desvíos, posiblemente para evitar que se reporten sin recaudos. La manera de calcularos es, de manera resumida: (Van Wieringen 2019)[7]

$$Var[\hat{\beta}(\lambda)] = \sigma^2((X^T X + \lambda I_{nxn}))^{-1} X^T X[(X^T X + \lambda I_{nxn})^{-1}]^T$$

Donde σ^2 es el error del modelo, X es la matriz de variables independientes de dimensión nxp, λ es el peso de la penalización e I es la matriz identidad de dimensión nxn.

Dado que nuestro modelo es sobre una muestra de la población utilizaremos una estimación $\hat{\sigma}^2$. Implementamos en R la función para calcular los desvíos estándar de nuestros coeficientes y actualizamos el ranking de los primeros diez jugadores con esta nueva información.

[6]Jelle Goeman, Rosa Meijer, Nimisha Chaturvedi (2018). L1 and L2 Penalized Regression Models. https://cran.r-project.org/web/packages/penalized/vignettes/penalized.pdf

[7]Wessel N. van Wieringen (2019). Lecture notes on ridge regression. https://arxiv.org/pdf/1509.09169;Lecture

playerid	player.firstName	player.lastName	team.abbreviation	coef	sd
9418	Joel	Embiid	PHI	2.173248	3.843356
9420	Robert	Covington	PHI	2.159759	2.741971
9402	Victor	Oladipo	IND	2.108312	3.295909
9265	Chris	Paul	HOU	2.019688	2.543393
9352	Eric	Gordon	HOU	1.932603	2.473872
9218	Stephen	Curry	GSW	1.791738	3.121359
9325	Giannis	Antetokounmpo	MIL	1.784740	2.602599
9152	Jimmy	Butler	MIN	1.776535	3.205942
9524	Otto	Porter Jr.	WAS	1.705704	3.048218
9224	Shaun	Livingston	GSW	1.651169	3.150640
9385	Andre	Roberson	OKL	1.637995	3.395044
9480	LaMarcus	Aldridge	SAS	1.613438	3.156169
10134	Jamal	Murray	DEN	1.551241	3.102419
9445	Damian	Lillard	POR	1.529906	3.126568
10124	Fred	VanVleet	TOR	1.524522	3.079476
9452	Al-Farouq	Aminu	POR	1.345331	3.229132
10087	Ben	Simmons	PHI	1.329176	3.164346
9490	Kyle	Lowry	TOR	1.323359	3.695157
9347	Tyus	Jones	MIN	1.303166	5.090437
13750	OG	Anunoby	TOR	1.256121	3.030928

A primera vista ya vemos que los desvíos estándar se sitúan alrededor de entre 2.5 y 3.5 puntos por cada 100 posesiones, mientras que el jugador con el mayor coeficiente aporta en promedio 2.17 puntos cada 100 posesiones. Los coeficientes están sesgados pero mantienen la propiedad de ser variables normales por lo que podemos aplicar los mismos tests que en una regresión lineal multivariada clásica.

```
# t-test a una cola
# Jugador rankeado 1
# Estan bien los DF?
tscore_1 = (player_ranking$coef[1] - 0)/player_ranking$sd[1]
pvalue_1 = 1 - pt(q = tscore_1, df = model$nobs - model$df)
# Jugador rankeado 20
# Estan bien los DF?
tscore_20 = (player_ranking$coef[20] - 0)/player_ranking$sd[20]
pvalue_20 = 1 - pt(q = tscore_20, df = model$nobs - model$df)
```

Realizando el test t para ver diferencias estadísticamente significativas respecto a 0 de los coeficientes podemos ver que tanto el jugador rankeado número uno “Embiid” como el jugador rankeado número veinte “Anunoby” tienen p-valores de 0.2858858y 0.3392807respectivamente, no encontrando evidencia suficiente para decir que tienen impacto positivo en los puntos de su equipo. Más allá de las limitaciones de nuestros estimadores, no poder asegurar estadísticamente que el jugador mejor rankeado aporta positivamente a su equipo parece un obstáculo importante.

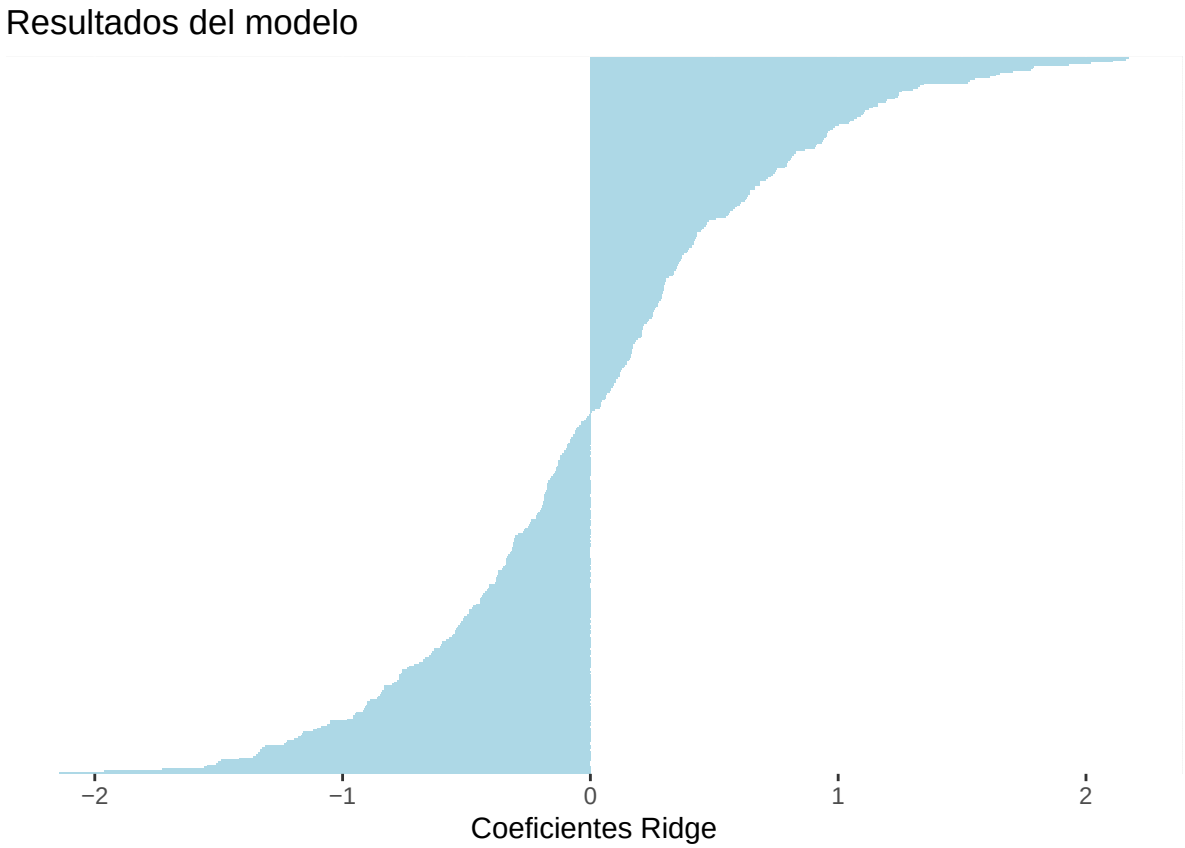
```
var_cov_ridge = readRDS("data/working/var_cov_ridge.RDS")
# Wald test. Coefficients differences.
# Estan bien los DF?
# Manually 1 vs 405
wald_1_20_init = player_ranking$coef[1] - player_ranking$coef[405]

sd_1_20 = sqrt(var_cov_ridge[paste0("x",player_ranking$playerid[1]),
               paste0("x",player_ranking$playerid[1])]+
               var_cov_ridge[paste0("x",player_ranking$playerid[405]),
               paste0("x",player_ranking$playerid[405])] -
               2*var_cov_ridge[paste0("x",player_ranking$playerid[1]),
```

```
              paste0("x",player_ranking$playerid[405])])

wald_1_20_t = wald_1_20_init / sd_1_20
pvalue_1_20 = 1 - pt(q = wald_1_20_t, df = model$nobs - model$df)
```

Yendo más allá, proponemos un test de Wald para comparar los coeficientes del rankeado número 1 contra el rankeado en la última posición. La hipótesis nula es que ambos coeficientes son iguales. El test de Wald se basa en que la variable aleatoria generada por la resta de ambos coeficientes sea igual a 0 en la hipótesis nula. Se la divide por su desvío estándar generando un estadístcio con distribución T de Student y luego se compara con las regiones críticas. En este caso el p-valor es de 0.209339, lo cual nos dice que de ser cierta la hipótesis nula encontrar un valor al menos tan extremo como este para el estadístico tiene 0.209339 de probabilidad. Ni con un umbral de 0.1 podemos rechazar la hipótesis nula y podemos decir que no hay evidencia estadística para diferenciar al primero del últiom jugador del ranking por sus coeficientes.
Al margen de lo ya aclarado de que los desvíos estándar están sesgados, resultados como estos no parecen darle demasiada entidad al modelo. Pareciera que la reducción de la varianza mediante la regresión Ridge, tal como se hizo el experimento, no alcanza para generar coeficientes estables.



Por otra parte luego sumamos los coeficientes de cada jugador por equipo y armamos un ranking de equipos conformado por esta agregación. Nuevamente los resultados son bastante creíbles, particularmente en la parte alta de la tabla donde los equipos que más ganaron en la temporada regular figuran en los primeros puestos. Hay alguna situación excepcional como Washington que figura ante último y se clasificó a la instancia de playoffs (posterior a la temporada regular) y por lo tanto su posición en la tabla no condice con su resultado real. Ver tabla 1.

Table 1: Ranking de Equipos

team.abbreviation	coef_total
TOR	7.4521032
GSW	6.2565058
UTA	6.1705909
HOU	5.5760234
BOS	3.5150124
OKL	2.7990024
POR	2.5279148
SAS	2.1930539
IND	2.0294579
PHI	1.6609936
DEN	1.4930613
MIN	1.3478828
DET	1.3387354
LAC	1.3297672
NOP	1.1667564
MIA	0.9158809
CLE	0.2932305
WAS	-0.0827118
MIL	-0.7560678
ORL	-0.7602037
LAL	-0.9506350
CHA	-2.4650446
PHX	-2.4683634
DAL	-2.4746261
NYK	-2.7617399
ATL	-3.7017177
BRO	-3.9949572
MEM	-4.3493298
SAC	-5.1988806
CHI	-7.1576671

Observaciones temporales

Lo que más tiempo demandó fue conseguir los datos crudos de la API y transformarlos para llegar al dataset final, teniendo en cuenta que los datos tenían errores como más de 5 jugadores en cancha por equipo, fechas de partidos que no coincidían entre tablas por formato en distinto huso horario, etc. Además es bastante volumen de datos y los chequeos intermedios no son tan fáciles.
Por otra parte, ya superado el obstáculo de construir el dataset parece que la regresión está medianamente orientada y da resultados provisorios en sintonía con la realidad. Más allá de lo consguido hasta el momento quedan ciertas tareas pendientes:

- Analizar la variabilidad de los coeficientes para ver qué tanto nos dice un ranking de ellos.
- Utilizar la misma metodología para agregar más datos de temporadas previas y darle más robustez a la regresión.
- Coneguir los datos de la temporada 2018-2019 para utilizarla de Test set.