Estudio para la mejor elección de tipo de pista en jugadores de la ATP realizado en pyspark

Carlos María Canut Domínguez

Big Data - UPV

Resumen

En el presente estudio se va a utilizar el conjunto de datos de los partidos realizados en la ATP desde el año 2010 hasta el 2021. Con esto, se realizará una serie de modelos de clasificación con los que predecir en base a las características de un jugador vencedor y de otro perdedor, además del ranking en la ATP de ambos y el tiempo de duración de partida en minutos, cuál sería el tipo de pista más idónea para que el resultado planteado ocurriera.

Indice

Descripción del Proyecto

Para este proyecto se ha seleccionado un conjunto de datos con todos los partidos jugados en la ATP con el cual se ha decidido utilizar los siguientes campos:

- Mano dominante del jugador (w_hand, l_hand)
- Altura del jugador (w_height, l_height)
- Edad del jugador (w_age, l_age)
- Posición en la ATP del jugador (w_rank, l_rank)
- Duración de la partida en minutos (minutes)
- Rango del torneo (tourney_level)
- Tipo de pista (Surface)

** Los campos con w_ se refieres a ganadores y con l_ a perdedores**

w_hand	l_hand	w_height	l_height	w_age	l_age	w_rank	l_rank	minutes	tourney_level	surface
R	R	196.0	183.0	24.2984257358	21.4291581109	20.0	251.0	69.0	A	Hard
L	R	180.0	173.0	24.1204654346	32.0492813142	105.0	63.0	113.0	A	Hard
R	L	188.0	198.0	27.3483915127	22.5434633812	7.0	134.0	81.0	A	Hard
L	L	180.0	185.0	24.1204654346	26.1409993155	105.0	81.0	136.0	A	Hard
R	R	185.0	193.0	31.1047227926	23.3429158111	12.0	13.0	61.0	A	Hard

summary	w_hand	l_hand	w_height	l_height	w_age	l_age	w_rank	l_rank	minutes	tourney_level	surface
count	2392	2392	2392	2392	2392	2392	2392	2392	2392	2392	2392
mean	null	null	186.75627090301003	186.02759197324414	27.82013968533068	28.135315299229884	49.09531772575251	72.02132107023411	107.86580267558529	null	null
stddev	null	null	7.340978466231814	7.098994660527587	3.583103124835897	3.6729781088988136	59.7837111896956	70.89356750866155	45.0833881529782	null	null
min	L	L	163.0	163.0	17.2320328542	17.9329226557	1.0	1.0	8.0	A	Clay
25%	null	null	183.0	183.0	25.1663244353	25.3826146475	14.0	30.0	79.0	null	null
50%	null	null	188.0	185.0	27.8576317591	28.205338809	34.0	58.0	100.0	null	null
75%	null	null	190.0	190.0	30.2422997947	30.6995208761	66.0	90.0	131.0	null	null
max	R	R	208.0	208.0	39.5811088296	39.9342915811	1042.0	836.0	1146.0	M	Hard

Tras decidir que datos se utilizarían, se realizó un muestreo estratificado del conjunto en base al año en el que se jugó el partido:

Datos totales: 19459

year	count
2016	1983
2012	2621
2020	268
2019	795
2017	1599
2014	2169
2013	2408
2018	1198
2011	2592
2021	120
2015	1093
2010	2613

Nuevos datos reducidos: 2392

year	count
2016	226
2012	334
2020	34
2019	88
2017	181
2014	248
2013	311
2018	162
2011	333
2021	17
2015	115
2010	343

Finalmente observamos el total de partidas tenemos por cada tipo de pista:

surface	count
Clay	729
Hard	1414
Grass	249

Podemos observar que falta la pista de alfombra, esto es porque al tener pocos registros y para los entrenamientos se decidió eliminarla.

Análisis de variables

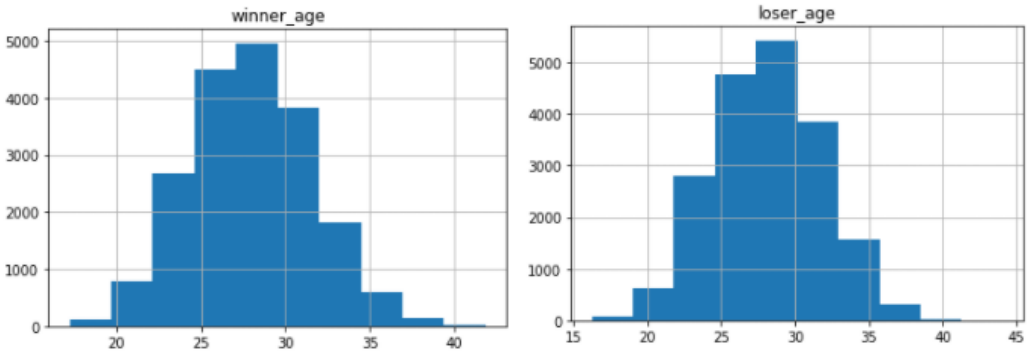
Se ha realizado un análisis variable a variable y también se han observado ciertas variables en relación a otras:

Edad:

resumen age

mean_winner	median_winner	Q1_winner	Q3_winner	kurtosis_winner	skewness_winner
27.9448683043329	27.9507186858	25.3360711841	30.410677618100003	-0.24807749405670076	0.11390401598089166

mean_loser	median_loser	Q1_loser	Q3_loser	kurtosis_loser	skewness_loser
28.098915218606994	28.1314168378	25.4592744695	30.6584531143	-0.25142836751162667	0.05736302014858706



Podemos observar la edad de los jugadores ganadores y de los perdedores se distribuye siguiendo una normal si miramos la asimetría, curtosis y los histogramas, también se puede observar que la edad media de los perdedores suele ser mayor.

surface	mean(winner_age)	mean(loser_age)	max(winner_age)	max(loser_age)	min(winner_age)	min(loser_age)
Carpet	24.25272468803077	27.978307797600003	34.9678302533	33.6755646817	18.8911704312	19.832991102
Clay	27.81269322819466	28.08200889591144	40.2409308693	43.8302532512	17.2320328542	16.2354551677
Hard	27.9150914512067	28.09501089963693	41.8507871321	44.0602327173	17.2429842574	17.2046543463
Grass	28.409907806723666	28.25490127169548	39.3401779603	39.3785078713	17.6262833676	17.6262833676

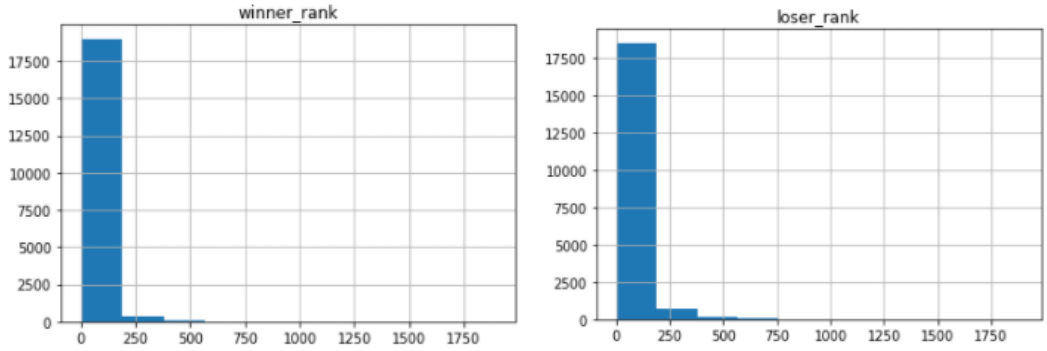
Aquí podemos observar como en las pistas de alfombra es donde la edad tiene un importante factor, ya que los jugadores más jóvenes tienen más posibilidad de ganar observando las medias.

Posición en ranking ATP:

resumen rank

mean_winner	median_winner	Q1_winner	Q3_winner	kurtosis_winner	skewness_winner
49.529626393956526	35.0	13.0	67.0	82.90235647372322	5.761769350311845

mean_loser	median_loser	Q1_loser	Q3_loser	kurtosis_loser	skewness_loser
73.75296777840587	57.0	31.0	90.0	73.92763102018864	6.260305163587122



En cuanto a la posición en el ranking de la ATP, tanto jugadores ganadores como perdedores tienen una posición alta en la ATP, siendo la de los ganadores superior como es coherente.

surface	mean(winner_rank)	mean(loser_rank)	max(winner_rank)	max(loser_rank)	min(winner_rank)	min(loser_rank)
Carpet	237.07692307692307	370.5833333333333	623.0	859.0	25.0	90.0
Clay	52.23704842428901	77.98320493066255	1112.0	1821.0	1.0	1.0
Hard	47.768067906224736	73.2530159501255	1890.0	1890.0	1.0	1.0
Grass	57.32218597063621	83.97755102040816	861.0	1109.0	1.0	1.0

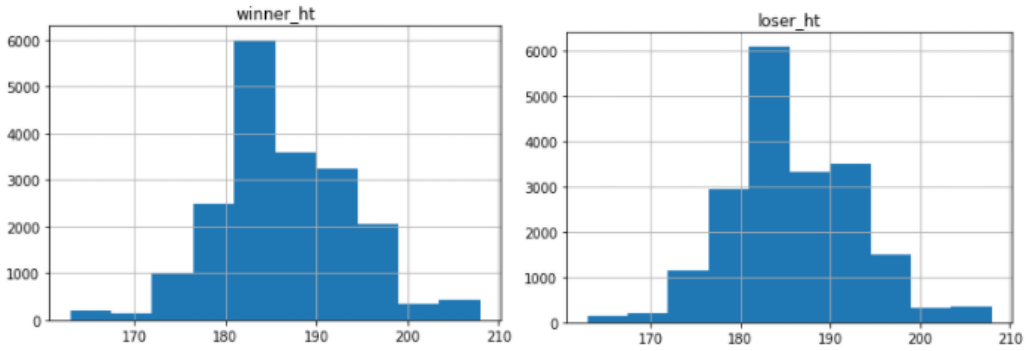
Tambien se puede observar que en pistas de tierra dura es donde más diferencia de rango podemos observar es en tierra batida.

Altura:

resumen height

mean_winner	median_winner	Q1_winner	Q3_winner	kurtosis_winner	skewness_winner
186.75101495451975	185.0	183.0	190.0	0.9442518611254185	0.12863910446872717

mean_loser	median_loser	Q1_loser	Q3_loser	kurtosis_loser	skewness_loser
186.06732103396885	185.0	183.0	190.0	0.9770046347996795	0.20134718123231332

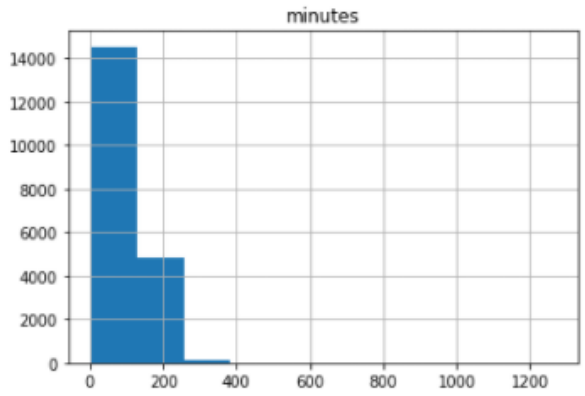


En cuanto a la altura de los jugadores, tanto los ganadores como los perdedores se distribuyen de la misma forma, además se aproxima a una normal.

Duración de la partida:

resumen game lenght

mean	median	Q1	Q3	kurtosis	skewness
108.41004162598284	101.0	78.0	131.0	47.511348984848155	2.804546105488834



Observando la duración de las partidas, podemos ver que la mayor parte de las partidas dura aproximadamente una hora y 40 minutos.

resumen game lenght by surface

surface	mean	median	Q1	Q3	kurtosis	skewness	count
Carpet	137.0	137.0	137.0	137.0	NaN	NaN	1
Clay	108.87576006573542	101.0	79.0	132.0	88.38749169746019	4.061482538610376	6085
Hard	107.25287663901526	99.0	78.0	129.0	34.538527193750575	2.370569473532891	11211
Grass	113.08649398704902	105.0	79.0	138.0	10.14761220251931	1.6882444171412594	2162

También se puede ver que las partidas más largas son las de césped, pudiendo deberse esto a que la pista de césped es la más rápida (ignorando alfombra ya que solo contamos con 1 registro).

Mano dominante:

winner_hand	count	percentage
L	2592	0.1332031450742587
R	16867	0.8667968549257413

loser_hand	count	percentage
L	2694	0.13844493550542167
R	16765	0.8615550644945783

Analizando el tipo de mano que utilizan los jugadores, podemos ver que en ambos casos, los diestros son la mayoría.

tourney_level_winner_hand	L	R
A	1473	9444
D	17	122
F	9	111
G	488	3355
M	605	3835

También es destacable el hecho de que independientemente de la categoría, se mantiene una diferencia significativa entre zurdos y diestros.

winner_rank_winner_hand	L	R
1.0	185	371
10.0	9	289
11.0	11	292
12.0	22	259
13.0	4	260
14.0	6	256
15.0	17	273
16.0	15	235
17.0	3	274
18.0	3	228
19.0	17	241
2.0	130	395
20.0	15	217
21.0	18	252
22.0	12	226
23.0	19	209
24.0	33	172
25.0	38	187
26.0	34	166
27.0	34	155

Cabe destacar que en los puestos 1 y 2, contamos con muchas partidas de jugadores zurdos a diferencia de puestos inferiores, este hecho puede que se deba a Rafael Nadal.

Descripción de los modelos:

Modelo de Regresión Logística con One-vs-All:

El clasificador OnevsRest (One-vs-All) es un tipo de clasificador multiclase que funciona con un clasificador binario de base, en nuestro caso una regresión logística.

El funcionamiento se fundamenta en obtener clasificaciones binarias entre todas las clases existentes, así con cada instancia comprueba si la clase es o no la que debería ser en la fase de entrenamiento.

Para hacer las predicciones, realiza una clasificación binaria con cada clase, y la que más confianza tenga es la que elige como resultado.

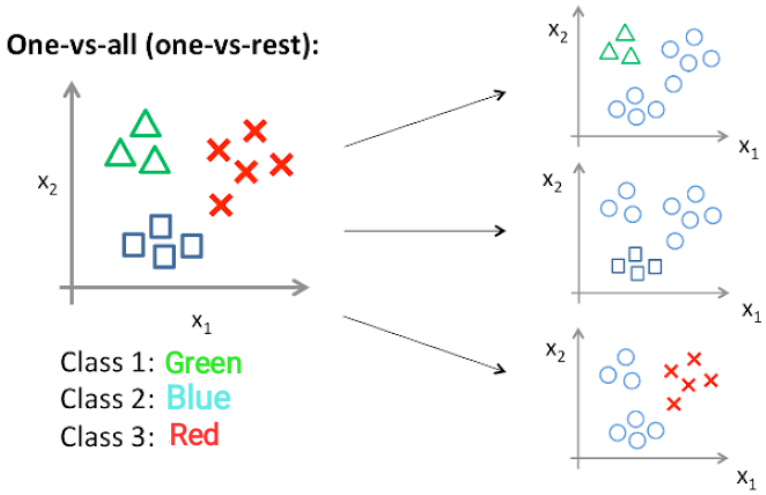


Ilustración One-vs-All. Anton Haugen (1 Marzo 2021).

Modelo de Arboles de decisión:

Los arboles de decisión funcionan en base a un primer nodo base del que se van generando distintas ramas en base a ciertas características que llevan a otros nodos, de manera que dependiendo de la instancia que estemos prediciendo, llegará hasta un nodo terminal que es el que indica el tipo de clase que se predice.

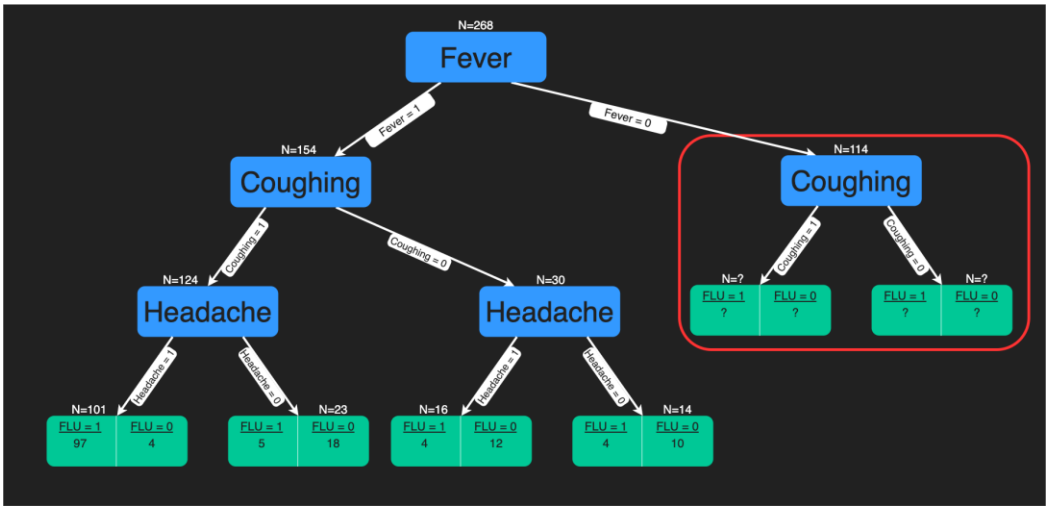


Ilustración Árbol de decisión. Casper Hansen (15 Septiembre 2021).

Modelo de Bosques Aleatorios:

Los Bosques aleatorios (Random Forest) hacen uso de los arboles de decisión, de manera que combinan diversos arboles de decisión, esto lo realizan creando cada arbol en base a un vector aleatorio de valores, todos los vectores se generan de manera aleatoria, y posteriormente se promedian entre estos, de manera que se evita el problema de overfitting.

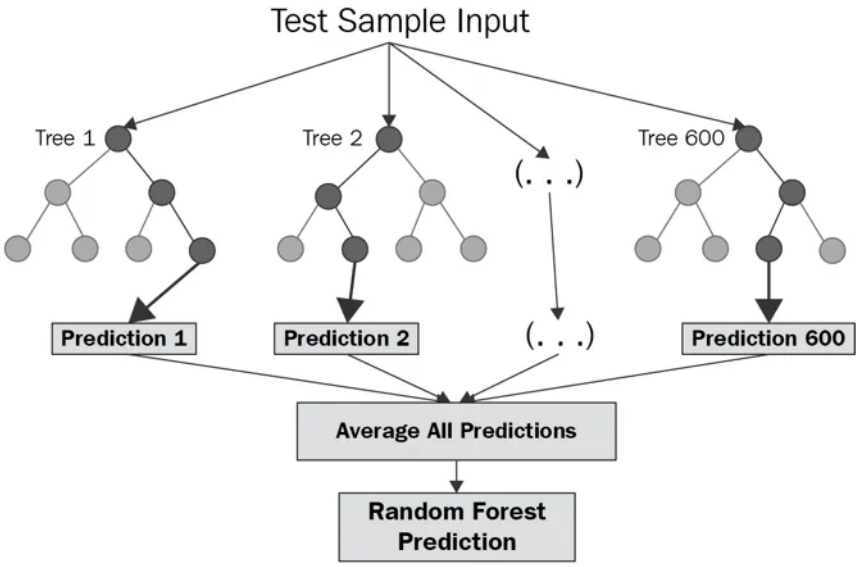
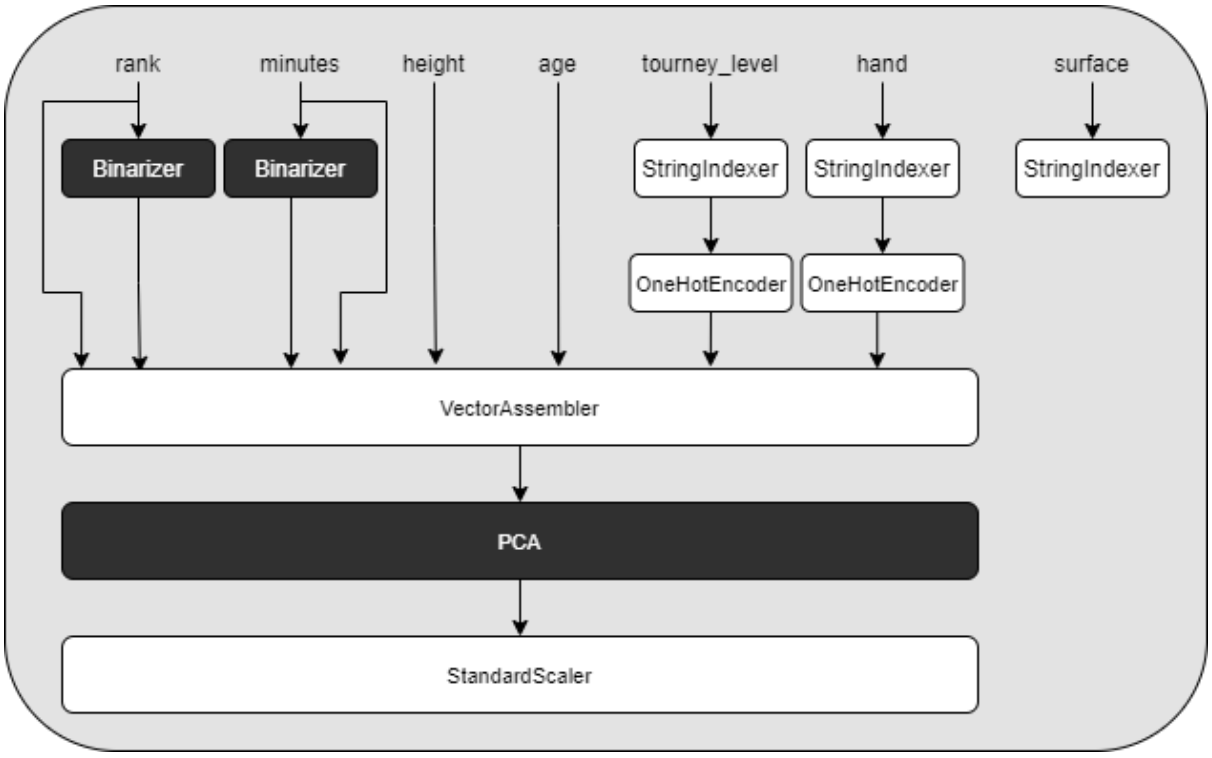


Ilustración Bosque Aleatorio. CFI (2021).

Descripción de la Pipeline:

Para la tubería se ha realizado una binarización diferenciando entre jugadores de posiciones más altas (observando la media de los jugadores ganadores) y también con el tiempo de duración medio de los partidos, posteriormente se ha juntado con todas las demás variables predictoras, se ha realizado un PCA con los valores más significativos y se han normalizado todos los valores para el análisis.



surface	surface_class	std_features
Clay	1.0	[0.2445262842576335,1.677924080540802,-0.6278985789006248,-1.5390143247204846,0.3208099598293323,1.1600979342983386,-1.248650186052327]
Hard	0.0	[0.05983408217682661,1.1777385826834315,-0.202784128266858,-1.5659437298755194,0.3102529820377186,0.5924697077162625,-0.725341609062653]
Clay	1.0	[-0.9102112307876484,-1.0153009539563096,0.1903123200629549,-0.5385002527554141,0.6811577913520477,0.8129224962828395,-1.1724352738879387]
Grass	2.0	[0.5917603512771668,-0.5604579789780375,0.6266299107065887,-0.36473809770870297,-0.5064932374976401,-0.441830468871907,-0.7508847513573038]
Clay	1.0	[-1.729273245789329,4.137027620742789,-1.6639289013868066,0.0749867876800114,-0.12498179811789945,-0.6061673273762095,-1.4587774178501327]

Este es el conjunto resultante después de aplicar la tubería.

Descripción de los hiperparametros:

Modelo de Regresión Logística con One-vs-All:

Los hiperparametros a editar de este modelo son:

- regParam `LogisticRegression.regParam, [0.01, 0.2]`
- elasticNetParam `LogisticRegression.elasticNetParam, [0.0, 0.2]`
- maxIter `LogisticRegression.maxIter, [10, 20, 30]`

Modelo de Arboles de decisión:

Los hiperparametros a editar de este modelo son:

- maxDepth `DecisionTreeClassifier.maxDepth, [4, 6, 12, 14]`
- maxBins `DecisionTreeClassifier.maxBins, [20, 24, 27, 29]`

Modelo de Bosques aleatorios:

Los hiperparametros a editar de este modelo son:

- maxDepth: `random_forest.maxDepth, [3, 4, 5, 6, 8]`
- maxBins `random_forest.maxBins, [22, 23, 24, 25, 26, 27, 29]`
- numTrees `random_forest.numTrees, [6, 7, 8, 10, 11]`

Descripción de las métricas para evaluar los modelos:

Para la evaluación de los modelos, se ha calculado el porcentaje de acierto del modelo, además de mostrar ciertas instancias de la predicción y como reacciona el modelo a estas.

Modelo de Regresión Logística con One-vs-All:

```
Error del modelo: 0.379888
Precisión mejor OnevsRest con Regresión Logística:  0.6201117318435754
```

surface	surface_class	prediction
Hard	0.0	0.0
Grass	2.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0

Modelo de Arboles de decisión:

```
Error del modelo: 0.379888
Precisión mejor Decision Tree:  0.6201117318435754
```

surface	surface_class	prediction
Hard	0.0	0.0
Grass	2.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0

Modelo de Bosques aleatorios:

```
Error del modelo: 0.377095
Precisión mejor Random Forest:  0.6229050279329609
```

surface	surface_class	prediction
Hard	0.0	0.0
Grass	2.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Clay	1.0	0.0
Hard	0.0	0.0

Análisis de los resultados:

```
Precisión mejor OnevsRest con Regresión Logística:  0.6201117318435754
Precisión mejor Decision Tree:  0.6201117318435754
Precisión mejor Random Forest:  0.6229050279329609
Precisión mejor Perceptron Multicapa:  0.61731843575419


['Random Forest', 0.6229050279329609]
['Decision Tree', 0.6201117318435754]
['OnevsRest', 0.6201117318435754]
['Perceptron Multicapa', 0.61731843575419]
```

Podemos concluir que aunque el modelo con más precisión sea el de Bosque aleatorio, al ser tan ligeras las diferencias entre los modelos, no se trata de una mejora sustancial.

Referencias

JeffSackmann (31 Mayo 2021). https://github.com/JeffSackmann/tennis_atp

Spark 2.3.0 (30 Mayo 2014). https://spark.apache.org/docs/2.3.0/ml-classification-regression.html

Spark 2.3.0 (30 Mayo 2014). https://spark.apache.org/docs/2.3.0/ml-pipeline.html

Spark 2.3.0 (30 Mayo 2014). https://spark.apache.org/docs/2.3.0/ml-features.html

Spark 2.3.0 (30 Mayo 2014). https://spark.apache.org/docs/2.3.0/ml-tuning.html

Wikipedia (14 Mayo 2021). https://es.wikipedia.org/wiki/Regresi%C3%B3n_log%C3%ADstica

Wikipedia (15 Febrero 2021). https://es.wikipedia.org/wiki/%C3%81rbol_de_decisi%C3%B3n

Wikipedia (5 Octubre 2020). https://es.wikipedia.org/wiki/Random_forest

Arya Mohapatra (25 Enero 2020). https://medium.com/analytics-vidhya/logistic-regression-from-scratch-multi-classification-with-onevsall-d5c2acf0c37c

CFI (2021). https://corporatefinanceinstitute.com/resources/knowledge/other/random-forest/

Casper Hansen (15 Septiembre 2021). https://mlfromscratch.com/decision-tree-classification/

Anton Haugen (1 Marzo 2021). https://antonhaugen.medium.com/introducing-mllibs-one-vs-rest-classifier-402eeab22493