Σχολη Επιστημης Υπολογιστων-Πανεπιστημιο Κρητης

CS543-Software Systems and Technologies for Big Data Applications

Εαρινο Εξαμηνο 2020-2021

Programming Assignment 2

Φοιτητής
Καλογερακης Στεφανος
ΑΜ:1205
Email:skalogerakis97@gmail.com

Διδάσκων
Χ. Κοζανιτης

Εισαγωγή

Στα πλαίσια της δεύτερης εργασίας του μαθήματος CS543 σκοπός ήταν τόσο η περαιτέρω εξοικείωση όσο η εμβάθυνση στις δυνατότητες του framework Apache Spark σε συνδυασμό με βασικές τεχνικές μηχανικής μάθησης. Πιο συγκεκριμένα, μελετήθηκαν ολόκληρη η διαδικασία απο την αρχική επεξεργασία ενός Dataset μέχρι τελικά την εφαρμογή τεχνικών όπως το Linear Regression, ενώ έγινε και χρήση της βιβλιοθήκης MLlib του Spark. Η άσκηση ήταν μια και ενιαία με διαφορετικές ενότητες να μελετάει κάτι διαφορετικό όπως θα δούμε και στην συνέχεια

Υλοποίηση

Παρακάτω μελετούνται αναλυτικά όλες οι ενότητες τις υλοποίησης ενώ στην επόμενη ενότητα υπάρχουν και οδηγίες εκτέλεσης

Οδηγίες εκτέλεσης

Η εκτέλεση κατά το development πραγματοποιήθηκε κατά κύριο λόγο στο περιβάλλον IntelliJ IDEA με την χρήση pom.xml αρχείο για την φόρτωση των απαραίτητων dependencies. Για την αποφυγή περαιτέρω προβλημάτων με τα dependencies παρακάτω παρατίθενται οδηγίες για την εκτέλεση του κώδικα μέσω spark shell σε κατάλληλα τροποποιημένο αρχείο.

- Μετάβαση στον φάκελο bin του directory του spark
- Εκτέλεση **./spark-shell -i <File_dir>/CodeSnippet.scala**

Στα τελικά παραδοτέα το αρχείο **CodeSnippet.scala** είναι το τροποποιημένο αρχείο που λειτουργεί με **spark shell**, ενώ το **CodeSnippet(INTELLIJ).scala** είναι το αρχείο που εκτελείται στο **IntelliJ**. Προκειμένου να εκτελείται επιτυχώς σε spark-shell έπρεπε να γίνουν απαραίτητες αλλαγες στον κώδικά μας(όπως την διαγραφή της main συνάρτησης) χωρίς να επηρεάζεται όμως η τελική λειτουργικότητα.

ΠΑΡΑΤΗΡΗΣΗ: Πραγματοποιήθηκαν κάποιες δοκιμές στο spark-shell και το αρχείο CodeSnippet.scala παράγει σωστά αποτελέσματα. Σε περίπτωση οποιασδήποτε ασυνέπειας με τα αποτελέσματα στις επόμενες ενότητες ή πρόβλημα εκτέλεσης παρακαλώ επικοινωνήστε μαζί μου για περισσότερες πληροφορίες σχετικά με το build στο περιβάλλον Intellij

1. Data Collection

1.2 - Load the raw data into spark

1.2.1

Sanity check count for data points(baseRDD): **4680**

1.2.2

Βλέπουμε παρακάτω τα top 5 datapoints με τον α/α(αύξων αριθμό) όπως εμφανίζονται στην κονσόλα

1) 1969,43.071036,-4.035391,23.572293,12.923576,-2.545036,5.052395,9.238151,-4.345975,5.224104,2.935664,-2.752638,1.729396

2) 1982,45.800256,41.148987,57.599295,5.695314,0.979893,-7.360076,-10.917191,-0.462272,-0.299410,-2.340378,0.261616,-2.427598

3) 2007,50.251554,27.845584,47.091303,11.080036,-43.505351,-17.997253,-5.284150,-11.754643,10.512851,2.192458,5.448426,1.704516

4) 1984,40.643545,6.281908,34.655208,-1.296938,-32.762731,-14.612497,7.706492,-8.353410,10.384000, -1.954814,-0.409230,-4.850200

5) 1986,45.747148,44.700684,22.545370,9.917018,10.745384,-13.228769,4.922118,-4.376980,20.309863,2.365600,1.039252,-2.439896

1.3 - Convert datapoints into LabeledPoints

1.3.3

Label of the first element: **1969.0**

1.3.4

Features of the first element: [43.071036,-4.035391,23.572293,12.923576,-2.545036,5.052395,9.238151,-4.345975,5.224104,2.935664,-2.752638,1.729396]

1.3.5

Length of the features of the first element: **12**

1.3.6

MAX: **2010.0**
MIN: **1926.0**

1.4 - Shift Labels

1.4.2

MAX: **84.0**
MIN: **0.0**

1.5 - Create Training, validation and test sets

1.5.3

TrainData count: **3745**
valData count: **459**
testData count: **476**

TOTAL count: **4680**

Εκτυπώνοντας τα επιμέρους count των dataset αλλά και το άθροισμα τους(Total count) παρατηρούμε ότι ισούται με την τιμή που είχαμε βρει στα πρώτα ερωτήματα και άρα δεν έχουν χαθεί τιμές

2. Create a Baseline Model

2.1 - The average label model

2.1.1

Average Shifted Song year: **71.29025367156208**

2.3 - Training, validation and test RMSE

2.3.2

RMSE for trainData: **11.665950068056636**
RMSE for valData: **12.14950697938246**
RMSE for testData: **11.3797257268283**

3. Linear Regression with Gradient Descent

3.3 - Implement Gradient Descent and Monitor Error Progress

3.3.2

Παρακάτω φαίνονται οι τιμές RMSE για κάθε Iteration στην αρχική υλποποίηση

Alpha Value: 2
Iteration RMSE: 327256.58670616645
Iteration RMSE: 1.1913615587521186E9
Iteration RMSE: 3.716749326073843E12
Iteration RMSE: 1.0329621191098386E16
Iteration RMSE: 2.6122725359622545E19
Iteration RMSE: 6.091765974656826E22
Iteration RMSE: 1.3227347053653002E26
Iteration RMSE: 2.6950915035397047E29
Iteration RMSE: 5.186086982746065E32
Iteration RMSE: 9.475987085536719E35
Iteration RMSE: 1.6516661830251467E39
Iteration RMSE: 2.7569817254659408E42
Iteration RMSE: 4.421992038840023E45
Iteration RMSE: 6.834928263159283E48
Iteration RMSE: 1.0206504736897878E52
Iteration RMSE: 1.4757275822187603E55
Iteration RMSE: 2.069984027313944E58
Iteration RMSE: 2.8216939499131805E61

Iteration RMSE: 3.7437349155477903E64
Iteration RMSE: 4.8412153760457073E67
Iteration RMSE: 6.109437440902957E70
Iteration RMSE: 7.532488701563253E73
Iteration RMSE: 9.082708801627587E76
Iteration RMSE: 1.072118801812537E80
Iteration RMSE: 1.239933806136456E83
Iteration RMSE: 1.4061446643720975E86
Iteration RMSE: 1.564800632733442E89
Iteration RMSE: 1.709951268170623E92
Iteration RMSE: 1.8360370439374304E95
Iteration RMSE: 1.9382536194744778E98
Iteration RMSE: 2.0128561967227864E101
Iteration RMSE: 2.0573778548053302E104
Iteration RMSE: 2.070745820064277E107
Iteration RMSE: 2.053291315291537E110
Iteration RMSE: 2.0066580805839793E113
Iteration RMSE: 1.9336267896981663E116
Iteration RMSE: 1.8378755580187137E119
Iteration RMSE: 1.7237031688141563E122
Iteration RMSE: 1.59574061671365E125
Iteration RMSE: 1.4586746994245522E128
Iteration RMSE: 1.317003005976055E131
Iteration RMSE: 1.174834075838581E134
Iteration RMSE: 1.0357405004574956E137
Iteration RMSE: 9.026670238912945E139
Iteration RMSE: 7.7789081935234E142
Iteration RMSE: 6.630274165160339E145
Iteration RMSE: 5.590733701420844E148
Iteration RMSE: 4.66475657536174E151
Iteration RMSE: Infinity
Iteration RMSE: Infinity

3.3.3

Αυτό που μπορούμε να παρατηρήσουμε απο το προηγούμενο ερώτημα είναι ότι το Gradient Descent δεν κάνει converge, αφού παρατηρούμε ότι φτάνει στο infinity και όχι σε κάποια πεπερασμένη τιμή.

3.3.4

Προκειμένου να παρατηρηθεί ποια απο τις alpha ή iteration number ευθύνεται και αποτρέπει το Gradient Descent να κάνει converge διενεργήθηκαν μια σειρά από δοκιμές τα αποτελέσματα των οποίων βλέπουμε συγκεντρωτικά στους παρακάτω πίνακες. Η τιμή στο πεδίο RMSE final Iter είναι η τελευταία τιμή που προέκυπτε απο το RMSE.

Για τιμή **alpha = 2**

Number of Iterations	RMSE final Iter
50	Infinity
200	NaN
10	9.475987085536719E35
1000	NaN

Για τιμή **alpha = 1**

Number of Iterations	RMSE final Iter
50	2.683675123996535E142
200	NaN
10	9.217982443400232E32
1000	NaN

Για τιμή **alpha = 0.1**

Number of Iterations	RMSE final Iter
50	1.2663377301193427E92
200	Infinity
10	8.592475411081726E22
1000	NaN

Για τιμή **alpha = 0.001**

Number of Iterations	RMSE final Iter
50	11.81379082130756
200	11.735586845850952
10	12.060936890721075
1000	11.677393311476166

Για τιμή **alpha = 0.01**

Number of Iterations	RMSE final Iter
50	3.1285056856957186E38
200	7.388345050940955E70
10	4.12454836890698E12
1000	4.643575134302941E53

Τα παραπάνω πειράματα φαίνονται με την σειρά που διενεργήθηκαν. Σαν αρχική παρατήρηση η μεταβλητή **alpha** είναι αυτή που φάνηκε ξεκάθαρα ακόμα και απο τα πρώτα πειράματα ότι επηρεάζει σε αρκετά μεγαλύτερο βαθμό την συμπεριφορά των πειραμάτων. Στα πρώτα πειράματα παρατηρείται λοιπόν μια καλύτερη τιμή RMSE όσο πέφτει η τιμή του alpha, για μεγαλύτερο αριθμό iteration όμως ακόμα η τιμή φτάνει στο infinity. Στο τέταρτο κατά σειρά πείραμα με τιμή **alpha=0.001** παρότι για πρώτη φορά για κανένα αριθμό iteration δεν βλέπουμε την τιμή infinity, το RMSE παραμένει υψηλό χωρίς μάλιστα να διαφοροποιείται σημαντικά για τους διαφορετικούς συνδυασμούς iterations. Αυτό είναι μια ένδειξη ότι το learning rate(alpha) είναι πολύ χαμηλό οπότε πρέπει να δοκιμαστούν μεγαλύτερες τιμές. Στο τελευταία λοιπόν πείραμα για **alpha=0.01** βλέπουμε ένα αρκετά καλό RMSE που το Gradient Descent γίνεται converge. Στα επόμενα ερωτήματα η τιμή alpha που έχει επιλεγεί είναι για **alpha=0.01** αφού όπως προαναφέρθηκε προκύπτει εν τέλει ενα καλό RMSE. Σαφώς υπάρχουν ακόμα περιθώρια και αρκετές δοκιμές που μπορούν να γίνουν προκειμένου να προκύψει το βέλτιστο learning step πάντως από τις δοκιμές η βέλτιστη τιμή αυτή θα βρίσκεται στο έυρος **0.01≤alpha<0.001**

3.3.5

Alpha Value: 0.01
Iterations: 50
Weights:
DenseVector(-3.1275841166968105E36, -4.5695614403650216E36, -1.2321925028121519E36, -3.076677014137528E35, 8.916579308230682E35, 6.881672205171658E35, -5.65819349491436E33, 1.8013568423591147E34, -3.746020497085776E35, -3.4015590940196965E35, 6.109499079883936E34, -1.3379067294633073E35)

Error train:
List(1568.8228731194592, 27116.575761775297, 397838.4441136834, 5155526.748222991, 6.0288065206357986E7, 6.448168075845886E8, 6.371805387544351E9, 5.865122146841678E10, 5.063750576895610 4E11, 4.12454836890698E12, 3.1851688952160254E13, 2.3418822660204016E14, 1.6452684426515228E15, 1.1078866256013762E16, 7.169881164831788E16, 4.4700495063401312E17, 2.6903031415652664E18, 1.5659658294766887E19, 8.830269346774385E19, 4.8308353521372647E20, 2.567504056137621E21, 1.3273064067407496E22, 6.8816927725961045E22, 3.2786920528205185E23, 1.5697174363452014E24, 7.338807526135178E24, 3.353213088503416E25, 1.4984836011682098E26, 6.553891365608792E26, 2.807268914085837E27, 1.1783365110850473E28, 4.8495537881788975E28, 1.958001522814033E29, 7.759274994713123E29, 3.019477421050409E30, 1.1543545418190118E31, 4.337377626974612E31, 1.602396590322856E32, 5.822812766185707E32, 2.081961668435662E33, 7.327218738832874E33, 2.539068769541882E34, 8.665914966978754E34, 2.9139941365708754E35, 9.656530723846144E35, 3.154498615659846E36, 1.016084587274218E37, 3.2279629486967405E37, 1.0116512153924534E38, 3.1285056856957186E38)

3.3 - Evaluate the model

RMSE on validation set = **3.0642093366694348E38**

4. Train using MLlib and grid search

4.1 - MLlib Linear Regression

4.1.1

Coefficients: [0.6339893978191026,-0.03739372681961374,-0.07761032962254956,0.1013753491568607, 0.025492863465037107,-0.19640391510199293,-0.04234490743521305,-0.07777325857561637, -0.15700510847017426,0.1764279332677829,-0.3522100459661825,0.011307722868484905]
Intercept: 43.03215805907352

4.1.2

RMSE on validation set: 11.31679149347788

4.1.3

Transformed validation set - First 10 predictions
COLUMN FORMAT: |**label**|**features**|**prediction**|
1) |56.0|[45.800256,41.148987,57.599295,5.695314,0.979893,-7.360076,-10.917191,-0.462272,-0.29941, -2.340378,0.261616,-2.427598]|68.12065929558337|

2) |58.0 |[40.643545,6.281908,34.655208,-1.296938,-32.762731,-14.612497,7.706492,-8.35341,10.384, -1.954814,-0.40923,-4.8502] |66.215893445572481
3) |38.0 |[44.763875,-17.45432,68.737722,-1.61186,-28.269173,7.171152,-39.716372,-5.467118,3.478588, -13.890788,4.407089,4.148463] |62.042188883893091
4) |74.0 |[43.174602,-1.595945,6.483995,25.442912,-17.185658,-13.020294,22.5405,2.705231,11.031711, 5.400918,2.67181,-6.328645] |71.702627411759581
5) |45.0 |[44.789073,17.241695,25.411734,3.625129,-36.286135,-3.65834,-4.730114,1.739752,3.658171, 0.95511,3.946162,8.749429] |67.340189551120661
6) |83.0 |[47.925393,-44.791807,33.470707,12.33591,31.53511,-22.129646,0.079541,9.620838,12.258473, -6.101398,6.270527,10.559479] |73.052584709623221
7) |75.0 |[40.17081,0.656904,57.552289,52.798073,19.098983,-0.975733,10.347024,-8.706106,17.054195, 5.369785,5.329147,6.516316] |66.745206503784421
8) |83.0 |[49.011671,17.117091,37.514203,-3.911994,-31.616632,-23.154532,6.953491,-0.85379,7.134362, 1.108201,0.574069,1.961562] |72.565870214718371
9) |83.0 |[43.366126,44.248099,-35.828968,32.348931,18.011108,-3.375495,-0.821978,8.16005,-8.795398, 14.599489,3.679698,6.752084] |78.1905952679478 |
10) |83.0 |[52.450455,59.96418,5.327727,-5.171187,-29.585206,-15.179119,-11.075841,-4.26492,3.003071, 10.708771,-3.624982,-5.744809]|78.762540923833981

4.2 - Grid Search

4.2.1

Η σύγκριση RMSE για διαφορετικές τιμές regularization parameter

Regularization Parameter	Value
0,1	11.31679149347788
1e-10	**11.314980241852886**
1e-5	11.314980374303529
1	11.349038490129756

4.2.2

Η regularization parameter που επιτυγχάνει το καλύτερο RMSE είναι 1e-10. Παρατηρούμε και εδώ ότι όσο ελαττώνουμε τιμές προκύπτει καλύτερο αποτέλεσμα αλλά από ένα σημείο και μετά δεν χρειάζεται να χαμηλσουμε άλλο την τιμή της συγκεκριμένης μεταβλητής με τις διαφορές να εξαιρετικά μικρές.

5. Add interactions between features

5.4 - Evaluate the interaction model on test data

5.4.1

New model RMSE: **11.007556504706073**
Baseline model RMSE: **12.14950697938246**

5.4.2

```
+——————+
| prediction|
+——————+
|74.34644597784165|
|75.77689112090543|
| 69.08373688808493|
|71.42529245238984|
|71.98687929877607|
|73.65763050684035|
|68.61348159481383|
|66.61507081871976|
|65.37188364582514|
|69.17264965456971|
|73.62175706422916|
|70.87769799985055|
|73.75768004954077|
|75.00423217710934|
|65.50409411661592|
|73.33193405887995|
| 66.0904785132354|
|76.57552609345151|
|75.12032299549274|
|67.57640068060502|
|61.42014343575739|
|67.47812384350466|
|66.01226310401178|
|70.89587182636421|
|69.54789578752552|
|79.40431176586377|
|73.08509422526382|
|76.04989880672173|
|75.73152259197951|
|75.44844884888724|
| 68.73130180018841|
|75.97346746134365|
|72.38229522159205|
|72.84862707374519|
|78.79910507848513|
|72.71857190495354|
| 66.3201583834054|
|76.23196948082358|
| 74.5045165430334|
|74.30359542168927|
|67.16171438442781|
|75.09616731579509|
|75.69945207187189|
```

```
| 77.7225638715151
|72.887701729271921
|75.485498023216181
|73.280467083038931
|69.603690778201131
|65.146227741703341
|65.744412646497791
+-----------------+
```

5.5 - Use pipeline to create the interaction model

RMSE final model test: **15.12192081231212**