

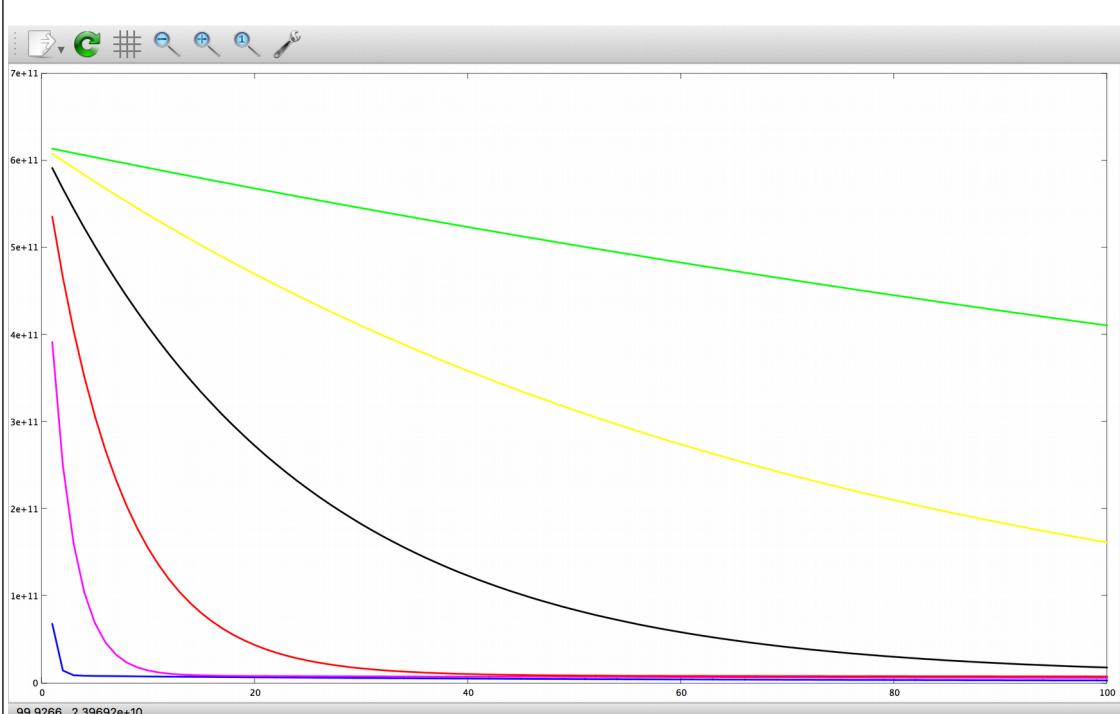
테스트 결과

- 작성자 : 이상욱, 장명규
- 일자 : 2017. 07. 20.

Ex1, 2, 3-1을 풀면서 Regression, Classification 프로그램 작성에 대한 연습을 하였고, UCI Online News Popularity Data 와 Bank Marketing Data 를 이용하여 테스트를 진행하였다. 각 feature들과 parameter들을 변형시키면서 결과값이 어떻게 변화하는지 눈으로 확인할 수 있었다.

News Data는 Linear Regression 테스트를 진행하였으며 Bank Data는 Multiclass Classification 테스트를 진행하였다.

1. Linear Regression - Learning Rate



<Linear Regression with Iteration 100>

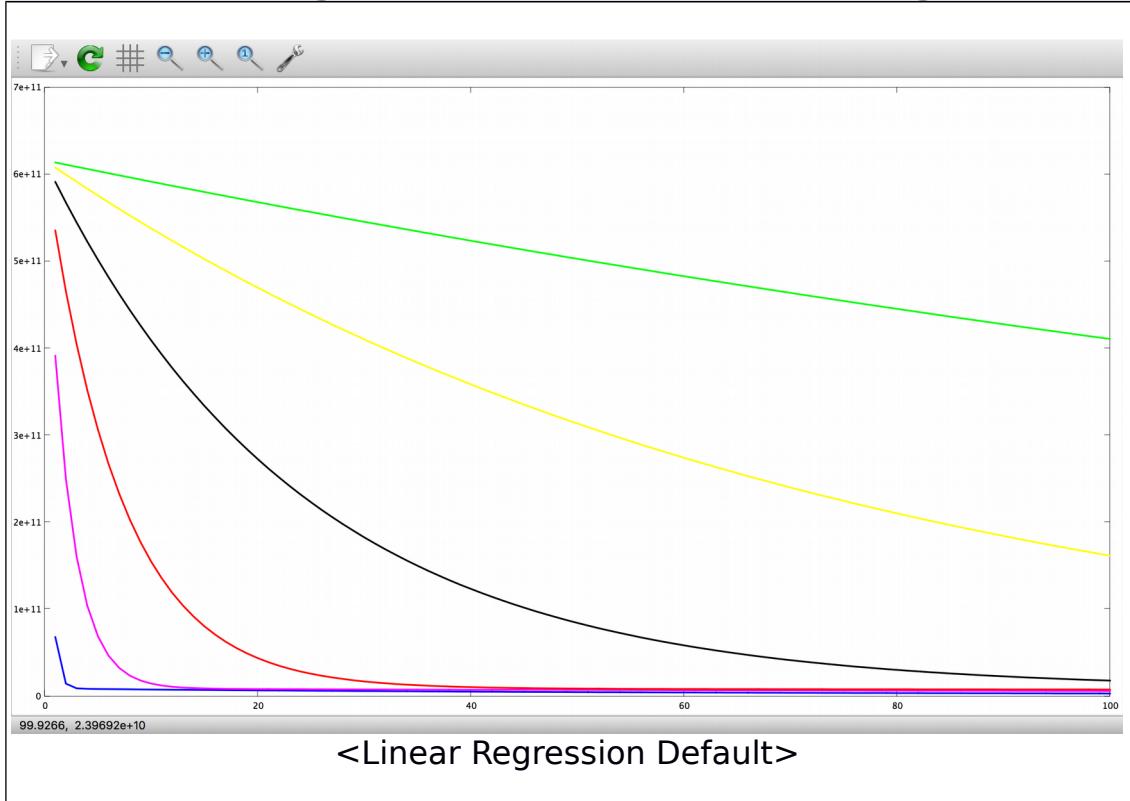
Learning Rate	1.0e-12	3.0e-13	1.0e-13	3.0e-14	1.0e-14	3.0e-15
Result	29838	52949	65986	178570	635030	997770

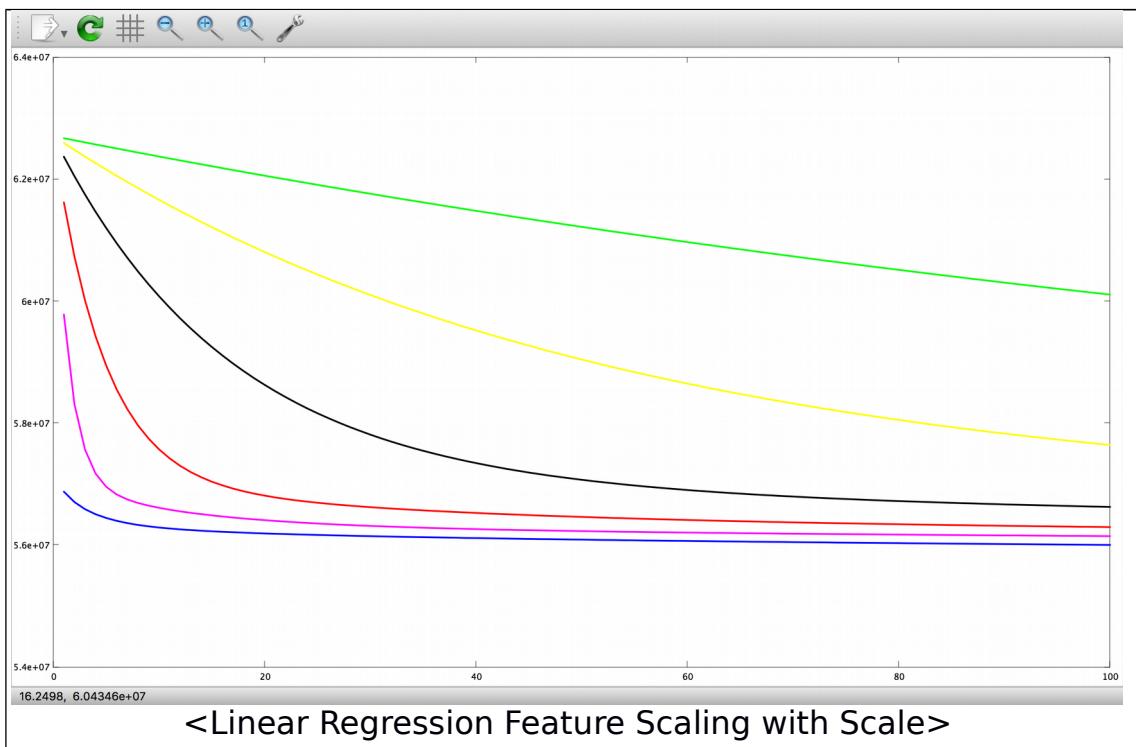
그래프 맨 아래부터 Learning Rate Alpha 를 1.0e-12 부터 3.0e-15 까지 Andrew Ng 강의에서 나온대로 3 배씩 감소시키며 Iteration 별 Cost 의 변화를 확인하였다.

Learning Rate 에 따라 Cost 의 감소율이 확연히 차이나는 것을 눈으로 확인할 수 있었다.

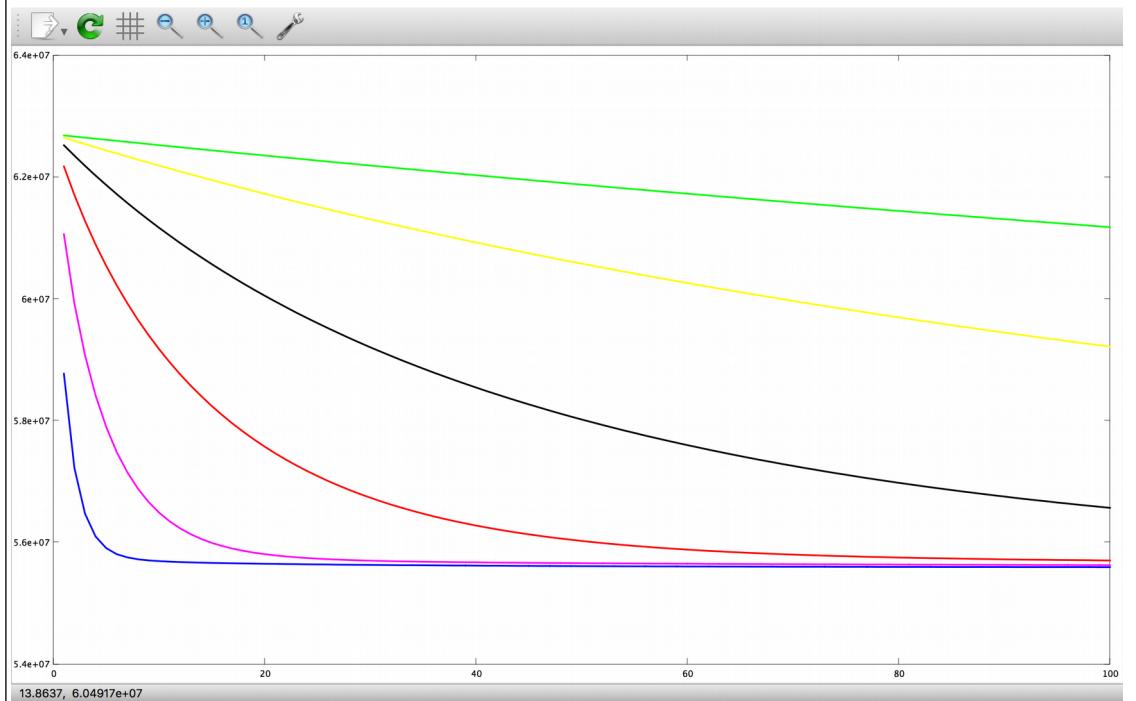
Result는 80%의 Train Set을 이용한 학습 결과를 실험하기 위하여 선정한 테스트 값으로, 10%의 Test Set을 이용하여 검증하였을 때, 평균 오차를 계산한 것이다.

2. Linear Regression - Feature Scaling





<Linear Regression Feature Scaling with Scale>



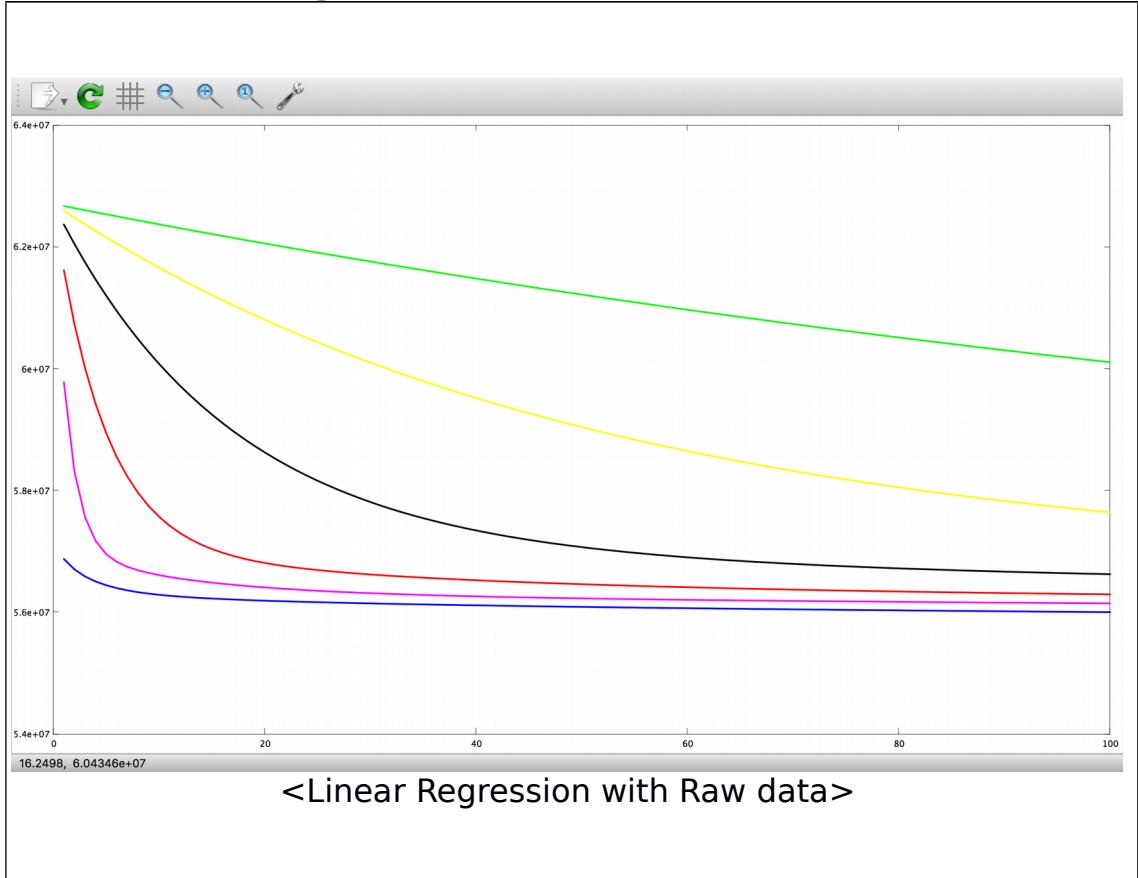
<Linear Regression Feature Scaling with Standard Deviation>

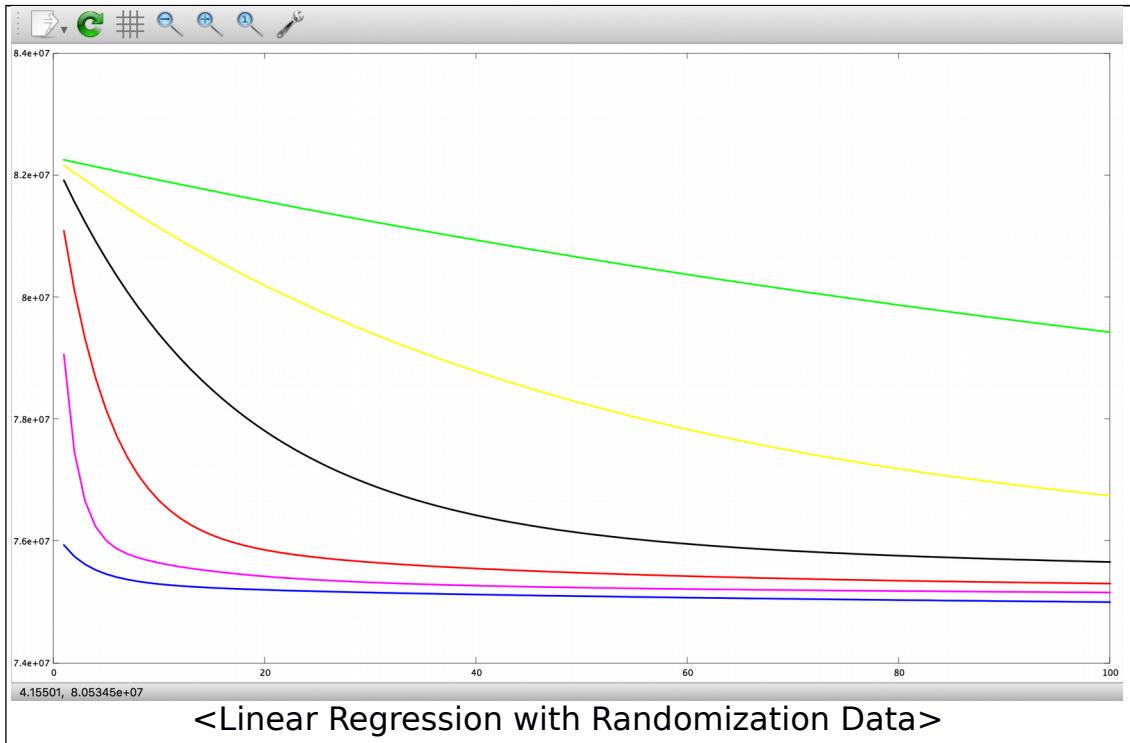
	Alpha	1.0e-12	3.0e-13	1.0e-13	3.0e-14	1.0e-14	3.0e-15
Default	Result	29838	52949	65986	178570	635030	997770
Scale	Alpha	1	0.3	0.1	0.03	0.01	0.003

	Result	2764.9	2809.3	2812.1	2693.3	2144.2	2110.6
STD	Alpha	0.3	0.1	0.03	0.01	0.003	0.001
	Result	2817.7	2807.7	2597.7	2053.1	2202.8	2633.0

Raw Data 의 y 값의 평균이 약 3300 인데, Default 를 이용하여 오차를 계산한 결과가 너무 차이가 커서, Data 에 Feature Scaling 작업을 진행하였다. 테스트 전에는 Scale 보다는 Standard Deviation 을 이용하였을 때 더 오차가 적어질 것이라고 생각하였으나, 테스트 결과는 Scale 을 이용하여 Feature Scaling 을 진행하였을 때 약간이나마 더 정확한 결과값을 얻을 수 있었다.

3. Linear Regression - Train Set Randomization





Raw Data	Alpha	1	0.3	0.1	0.03	0.01	0.003
	Result	2764.9	2809.3	2812.1	2693.3	2144.2	2110.6
Random Data	Alpha	1	0.3	0.1	0.03	0.01	0.003
	Result	3260.4	3262.4	3267.8	3202.4	2729.9	2795.6

테스트 도중 Raw Data 들을 눈으로 일부 직접 확인하였는데, 비슷한 Feature 값을 가진 데이터들끼리 모여있는 현상을 발견할 수 있었다. 혹시 비슷한 값을 가진 데이터들만 학습에 반영되고 상이한 데이터들을 이용해 Test를 진행하면 결과값이 바르지 않게 나올 수 있다고 판단하여 Data 들을 무작위로 다시 Random Sorting 하는 작업을 진행하였다.

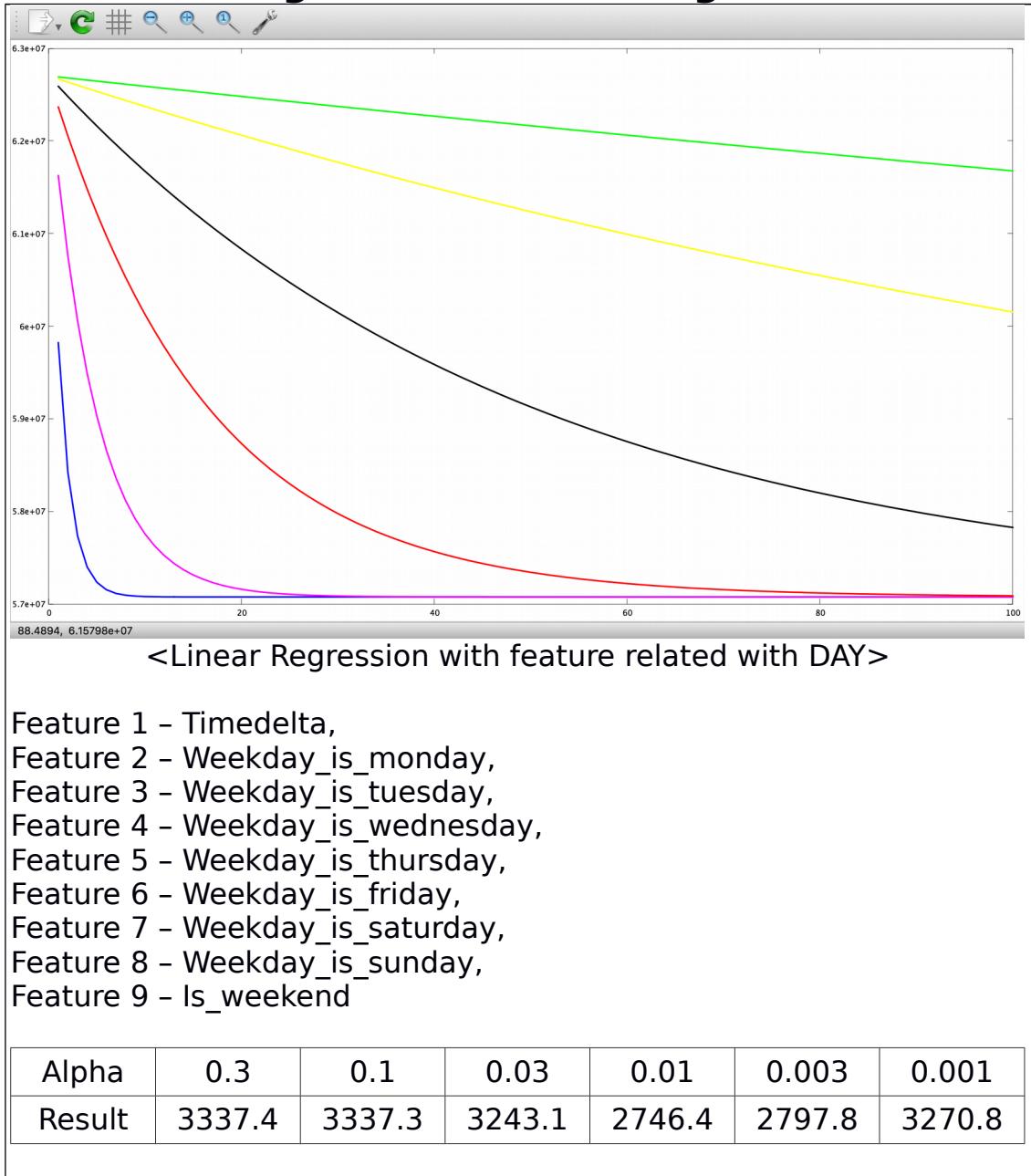
테스트 결과는 크게 차이를 발견할 수 없었으며 오히려 약간 Error 가 증가하는 현상이 발생하였다.

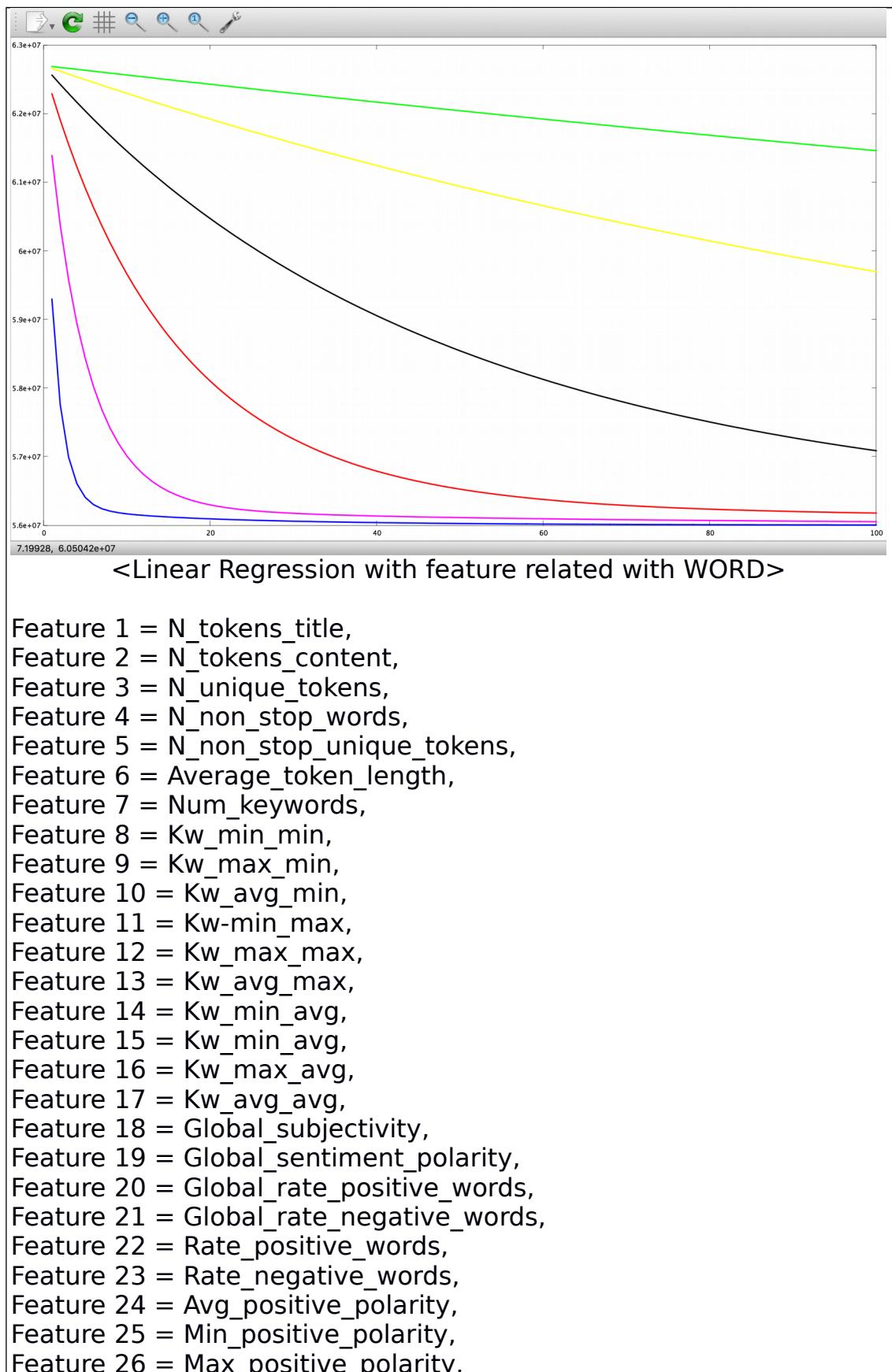
Res default = 31953, 55950, 68808, 132570, 530110, 859090(에러 평균) (1×10^{-12} , 3 10-13, 1 10-13, 3 10-14, 1 10-14, 3 10-15)

Res Scale = 3260.4, 3262.4, 3267.8, 3202.4, 2729.9, 2795.6
(1, 0.3, 0.1, 0.03, 0.01, 0.003)

Res std = 3325.2, 3320.7, 3223.2, 2721.0, 2794.7, 3270.1
(0.3, 0.1, 0.03, 0.01, 0.003, 0.001)

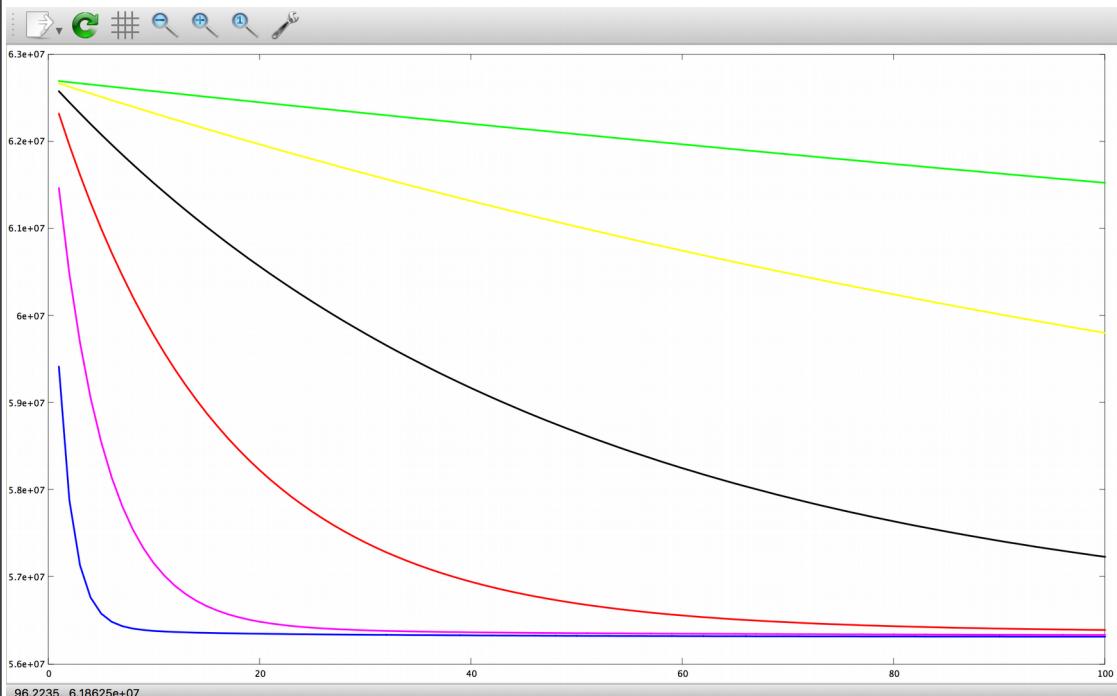
4. Linear Regression - Selecting Feature





Feature 27 = Avg_negative_polarity,
 Feature 28 = Min_negative_polarity,
 Feature 29 = Max_negative_polarity

Alpha	0.3	0.1	0.03	0.01	0.003	0.001
Result	3273.5	3279.2	3194.9	2743.4	2862.7	3274.5

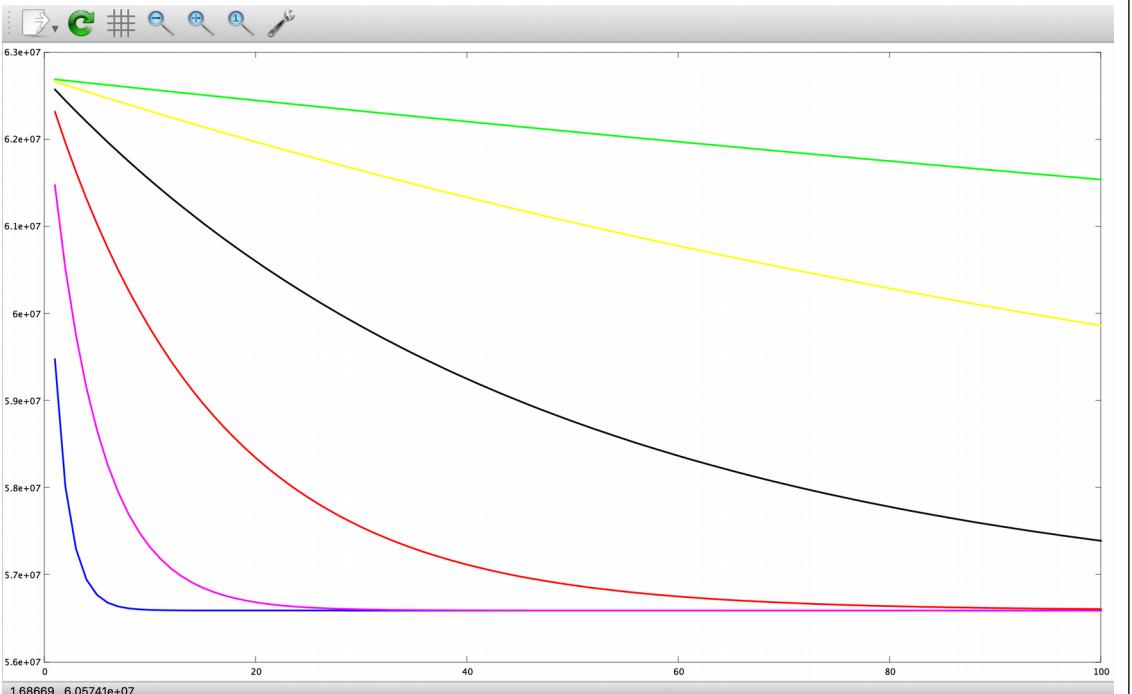


<Linear Regression with feature related with GENRE>

Feature 1 = Data_channel_is_lifestyle,
 Feature 2 = Data_channel_is_entertainment,
 Feature 3 = Data_channel_is_bus,
 Feature 4 = Data_channel_is_socmed,
 Feature 5 = Data_channel_is_tech,
 Feature 6 = Data_channel_is_world,
 Feature 7 = LDA_00,
 Feature 8 = LDA_01,
 Feature 9 = LDA_02,
 Feature 10 = LDA_03,
 Feature 11 = LDA_04,
 Feature 12 = Title_subjectivity,
 Feature 13 = Title_sentiment_polarity,
 Feature 14 = Abs_title_subjectivity,
 Feature 15 = Abs_title_sentiment_polarity

Alpha	0.3	0.1	0.03	0.01	0.003	0.001
-------	-----	-----	------	------	-------	-------

Result	3284.2	3284.8	3194.0	2742.5	2857.0	3271.9
--------	--------	--------	--------	--------	--------	--------

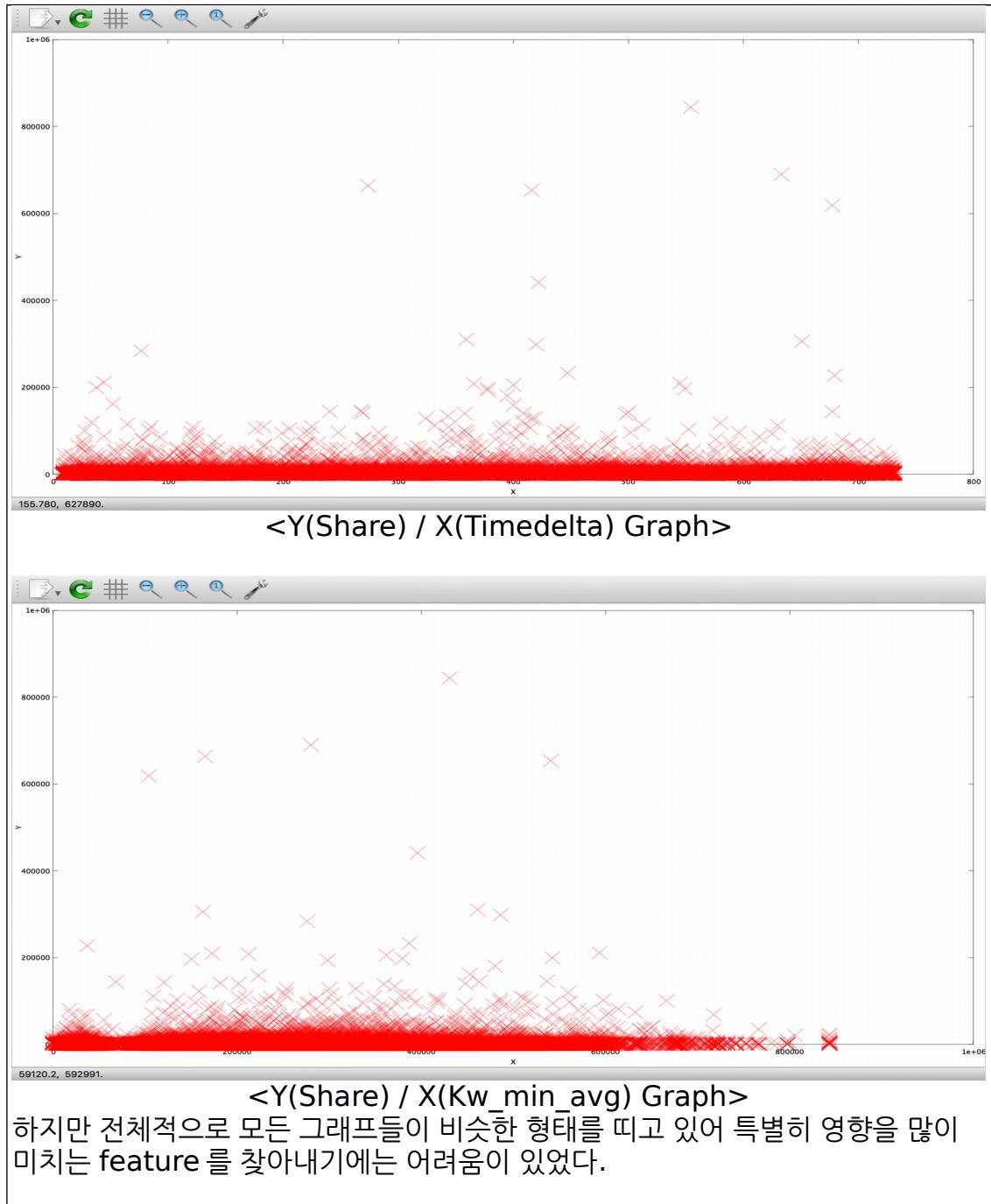


Feature 1 = Num_refs,
 Feature 2 = Num_self_refs,
 Feature 3 = Num_imgs,
 Feature 4 = Num_videos,
 Feature 5 = Self_reference_min_shares,
 Feature 6 = Self_reference_max_shares,
 Feature 7 = Self_reference_avg_shares

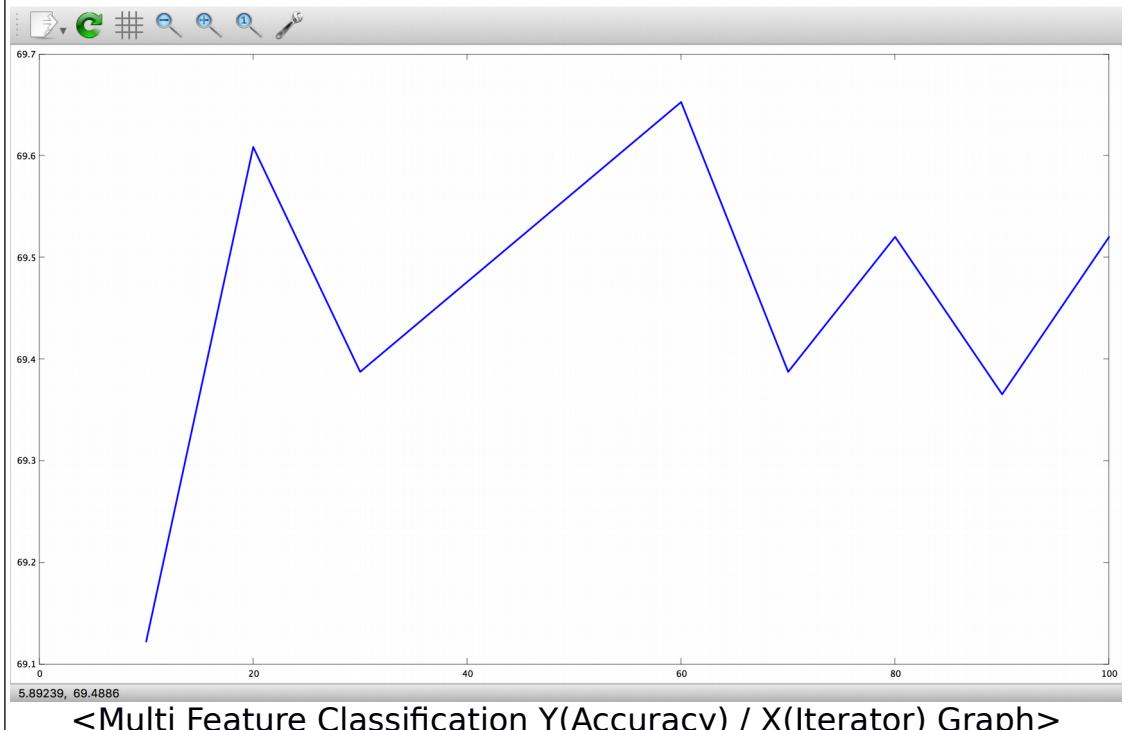
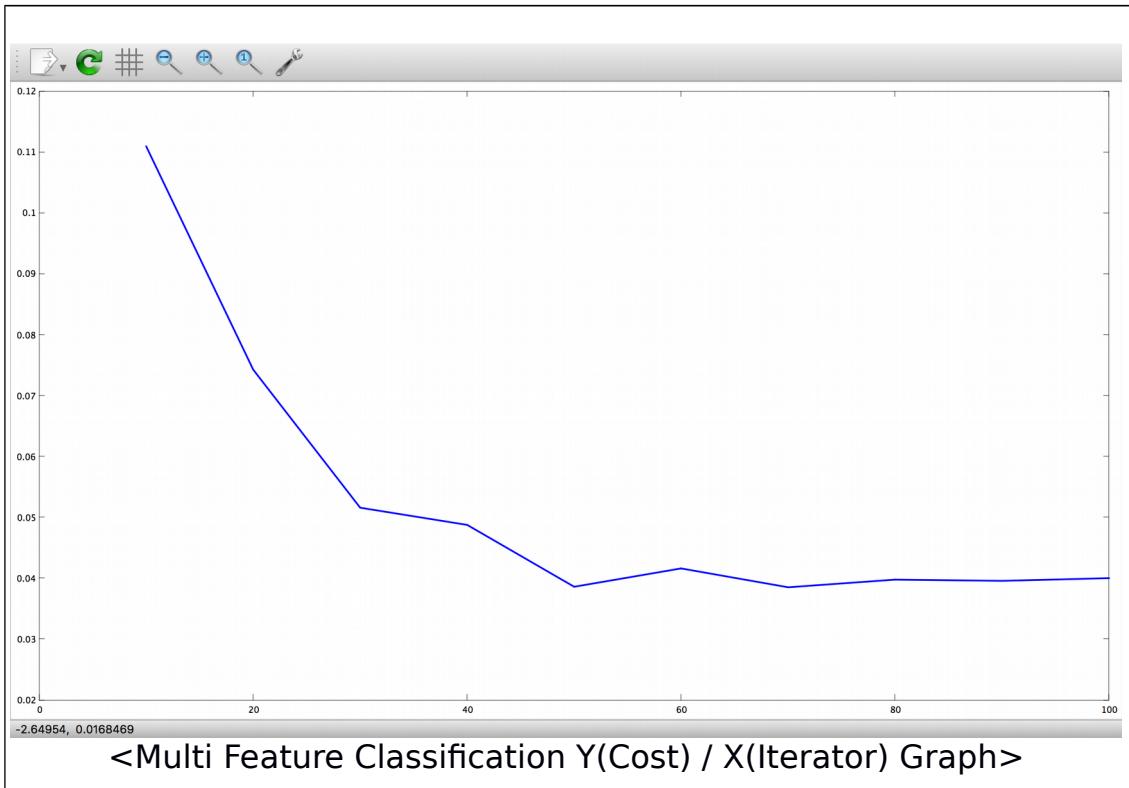
Alpha	0.3	0.1	0.03	0.01	0.003	0.001
Result	3336.6	3336.4	3242.0	2757.4	2830.5	3273.3

59 개의 feature들을 모두 사용하여 Linear Regression을 진행하였을 때 발생하는 Error가 너무 크다고 생각하였다. 그래서 영향을 많이 미치지 않는 feature들을 제외시키고 Regression을 진행해 보기 위하여 크게 4 가지(날짜, 단어, 장르, 미디어) 분야로 나누어 그 결과를 확인해 보았다. 하지만 4 개의 분야들 모두 기존 Error에서 크게 변하지 않는 결과를 확인할 수 있었다.

그래서 각각의 Feature들이 어떤 상관관계를 가지고 있는지 y 와 1:1 로 그래프를 그려서 확인해 보았다.



5. Classification - Iteration



Iteration	10	20	30	40	50
Cost	0.1109	0.0742	0.0515	0.0487	0.0385
Accuracy	69.122	69.608	69.387	69.476	69.564

Iteration	60	70	80	90	100
Cost	0.0415	0.0384	0.0397	0.0395	0.0399
Accuracy	69.653	69.387	69.520	69.365	69.520

반복문의 수와 cost, accuracy 의 상관관계를 확인하기 위해 반복문의 횟수를 10 번씩 증가시키며 확인하였다. Cost는 Linear Regression 과 같이 일정 수준으로 떨어진 이후에는 비슷한 수준을 유지하는 것을 확인할 수 있었으며, Accuracy 또한 일정 수준에 도달한 이후에는 약간씩 상향 / 하향 되었으나 비슷한 수준을 유지하는 것을 확인할 수 있었다.

6. Classification - lambda

