실험 보고서

- 작 성 자 : 오근택, 김정민
- 일 자 : 2017. 07. 20.

1. 실험 목표

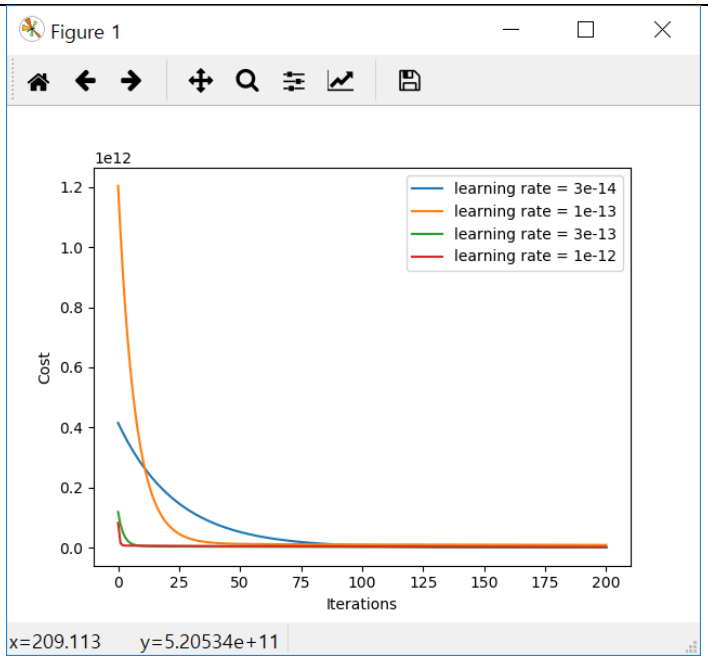
실제 데이터를 가지고 지금까지 배웠던 Machine Learning Algorithm 을 적용하고 궁금했던 부분을 실험한다. 실험 데이터는 UCI Machine Learning Repository 에서 Online News Popularity Data Set 을 사용한다. Non-predictive 한 feature 2 개를 제외한 58 개의 feature 를 사용하여, 온라인 뉴스 기사의 공유 횟수(share)를 예측 할 예정이다. Sample data 는 39797 개이다.

2. 실험 내용

코드 구현은 python 과 tensorflow 를 이용하였다. Feature 가 58 개 이므로 Linear Regression with multiple variable 모델을 구현하였다. 학습 회수는 200 번으로 설정하였으며, 결과에 따라서 learning rate 를 조정하였다. 또한, 데이터 셋에서 임의로 training data, validation data, test data 를 각각 80%, 10%, 10%로 분류하였다. Cost 는 Mean Square Error 함수를 이용하였으며, 학습 방법으로 Gradient Descent 를 사용하였다.

1) Linear Regression with multiple variable

기본적인 Linear Regression with multiple variable 을 학습한 결과는 다음과 같다.



Learning rate 별 cost-iterations 그래프의 결과

```
In learning rate : 1e-12
Validation Cost : 222365296.000000, Test Cost : 302703424.000000
```

최적의 learning rate 에 대한 validation cost, test cost 의 값

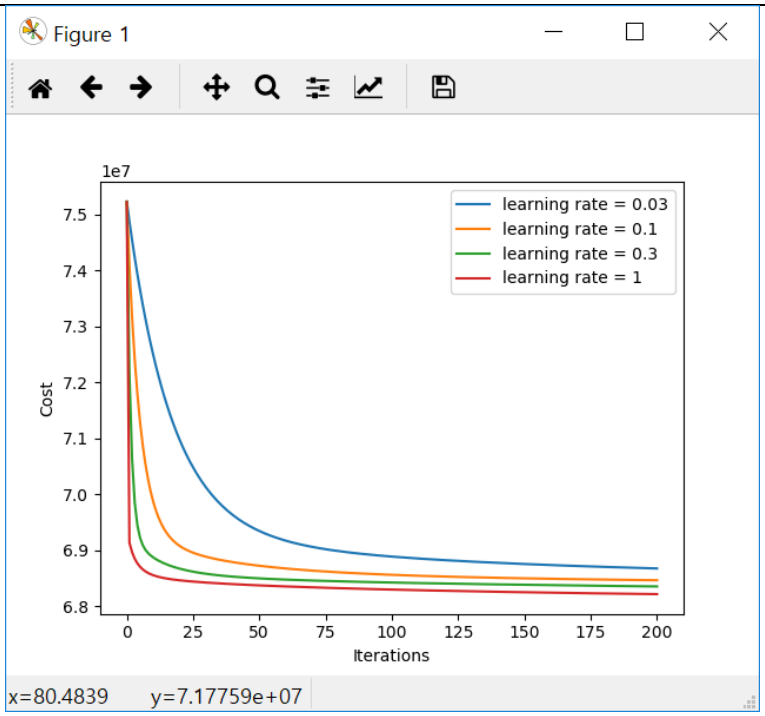
그래프에서 볼 수 있듯이 최적의 learning rate 는 1e-12 였으며, 이 코스트에 대한 validation cost 와 test cost 는 각각 약 $2.2 * 10^8$, $3.0 * 10^8$ 이었다. Learning rate 는 이 이상이 될 시 무한으로 수렴한다. Cost 그래프의 모양은 최적의 모양처럼 보였지만, 최적의 learning rate 가 너무 낮기 때문에 feature 간 range 차이가 많이 날 것이라고 판단하여 feature scaling 을 하였다.

2) Normalization

Feature range 와 feature mean 을 이용하여 normalization 을 해주었다. Normalization 의 식은 다음과 같다.

$$x_i := \frac{x_i - \mu_i}{s_i}$$

μ_i는 각 feature 의 평균이고, s_i 는 각 feature 의 range(max - min)이다. Normalization 한 feature 를 이용하여 학습한 결과는 다음과 같다.



Learning rate 별 cost-iterations 그래프의 결과

```
In learning rate : 1.0
Validation Cost : 33278152.000000, Test Cost : 85237800.000000
```

최적의 learning rate 에 대한 validation cost, test cost 의 값

최적의 learning rate 는 1.0 으로 전 실험에 비해 확연이 값이 증가한 것을 알 수 있다. Validation cost 와 test cost 도 각각 약 $3.3 * 10^7$, $8.5 * 10^7$ 으로 각각 7 배, 3 배 이상 줄었다. 이를 통하여, normalization 을 하면 값의 range 가 정규화가 되고 range 가 정규화 함에 따라, learning rate 의 제약이 크게 없어짐을 알 수 있었다. 또한, Learning rate 제약의 사라짐은 cost 가 보다 적은 값으로 수렴 할 수 있다는 사실을 내포하였다.

3) Polynomial Linear Regression with regularization

Cost 가 아직 너무 크고, regularization 을 적용해 보기 위하여, Polynomial Linear Regression 을 구현하려고 하였다. 하지만, feature 의 수가 58 개이므로, 2 차식으로 만들면 항의 개수가 약 1200 개가 되기 때문에 제한된 시간내에 현실적으로 구현하기가 어렵다고 판단하였다. 대신, 각 feature 간 조합된 항을 제외한 feature 각각의 제곱 값을 multiple variable Linear Regression 에 추가하였다. 우리가 만든 Polynomial Linear Regression 식은 다음과 같다.

$$H_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \dots + \theta_{58} x_{58} + \theta_{59} x_1^2 + \theta_{60} x_2^2 + \dots + \theta_{116} x_{58}^2$$

또한, Polynomial Linear Regression 에 대하여 Regularization 을 적용하기 위하여, 다음과 같은 Regularization 을 적용하였다. λ 는 0, 0.01, 0.1, 1 을 적용하였다.

$$\theta_j := \theta_j - \alpha \left[\left(\frac{1}{m} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)}) x_j^{(i)} \right) + \frac{\lambda}{m} \theta_j \right] \qquad j \in \{1, 2 \ldots n\}$$

Regularization 을 적용한 Polynomial Linear Regression 의 결과는 다음과 같다.

```
Validation Cost : 33237598.000000, Test Cost : 85294952.000000
```

λ = 0 일때, validation cost, test cost 의 값

```
Validation Cost : 33243420.000000, Test Cost : 85304816.000000
```

λ = 0.01 일때, validation cost, test cost 의 값

```
Validation Cost : 33286974.000000, Test Cost : 85381152.000000
```

λ = 0.1 일때, validation cost, test cost 의 값

```
Validation Cost : 33432634.000000, Test Cost : 85677888.000000
```

λ = 1 일때, validation cost, test cost 의 값

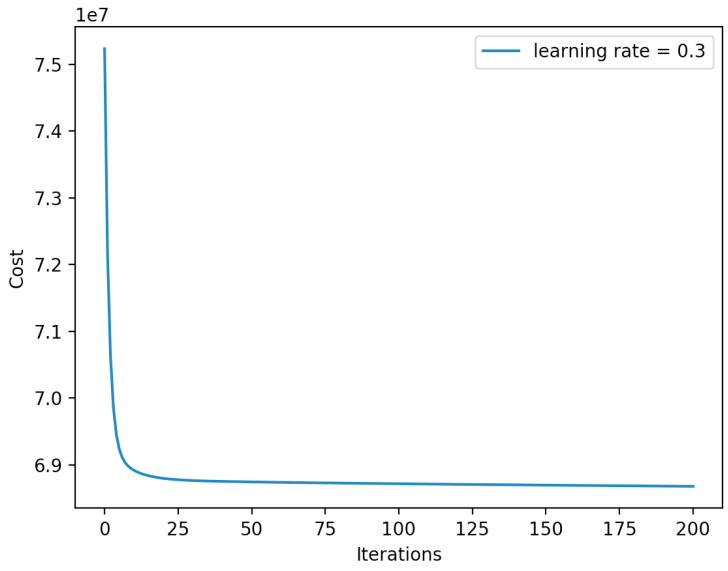
Polynomial Linear Regression 을 적용하였을 때 cost 를 더 낮출 수 있었지만 값의 변화는 미미했다. 2 차식의 항의 개수를 원래 개수의 약 1/10 인 117 개로 제한하였기 때문이라고 판단한다. 하지만, λ 값의 증가에 따라 cost 가 미미하지만 증가하는 것을 관찰함으로써, 1 차 Linear Regression 과 같은 형태로 변한다는 것을 알 수 있었다.

4) Linear Regression with high valued theta

Normalization 한 모델에서 theta 의 절대값이 모델에 영향을 미치는 정도와 연관성을 확인하고 싶었다. Normalization 을 하여서 X 의 값이 feature 마다 비슷하다면 theta 값에 따라서 결과값인 y 가 영향을 받기 때문에 theta 의 절대값이 큰 양수이면 결과값에 긍정적으로 크게 영향을 끼칠 것이고, theta 의 절대값이 큰 음수이면 결과값에 부정적으로 크게 영향을 끼칠 것이라는 가설을 세웠다. 그리고 실험을 하기 위해서 위에서 한 실험을 바탕으로 나온 theta 값으로 csv 파일과 비교해보았다. 그 중 theta 의 절대값이 1 보다 큰 feature 들을 추출하였다.

[1.08542979]	num_self_hrefs
[1.17844069]	num_videos
[-1.95446277]	kw_max_min
[1.46818912]	kw_avg_min
[-2.45708847]	kw_avg_max
[1.47399783]	kw_avg_avg
[-1.80516708]	self_reference_max_shares
[1.28917885]	weekday_is_tuesday
[1.07196486]	weekday_is_wednesday
[-3.05940866]	LDA_00
[3.86754298]	LDA_01
[-1.01007187]	LDA_02
[-2.47178102]	LDA_03
[1.48454213]	global_subjectivity
[-1.17790723]	global_sentiment_polarity
[2.66706848]	rate_negative_words
[-1.08454084]	min_positive_polarity
[2.34135199]	max_positive_polarity
[-1.00680053]	avg_negative_polarity
[-1.14028037]	min_negative_polarity
[-1.67315841]	title_subjectivity
[1.15427446]	abs_title_subjectivity

이 feature 들이 결과 값에 큰 영향을 끼칠 것이라는 기대를 하며 원래 58 개의 feature 중 22 개의 feature 만을 사용하여 다시 linear regression 모델을 세우고 학습하였다. 그 결과는 다음과 같다.



cost-iterations 그래프의 결과

```
In learning rate : 0.3
 Validation Cost : 33512136.000000, Test Cost : 85801176.000000
```

최적의 learning rate 에 대한 validation cost, test cost 의 값

위 그림과 같이 중요한 feature 만을 이용하여서 h(x)을 세우면 cost 값이 내려갈 것이라고 기대하였지만 결과는 큰 차이가 없고 유사하다. 따라서 theta 값이 작더라도 그 feature 는 결과값 y 에게 영향을 미치므로 이를 무시하면 안되는 것을 알 수 있었다.

5) Accuracy and conclusion

Training set, validation set, test set 에 있는 각각 임의의 5 개 데이터로 각 모델의 정확도를 실험하여 비교했다. 실험 결과는 다음과 같다. abs 는 예측 값과 실제 값 간의 차이이다. Cost 는 mean square error 이다.

Linear Regression without normalization

training data set			validataion data set			test data set		
y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)
16100	-14213.617	30313.6172	2500	34393.6211	31893.6211	826	4350.6972	3524.6972
508	-1780.3219	2288.3219	1300	6586.646	5286.646	920	2106.9656	1186.9656
1300	-8851.8467	10151.8467	559	2113.564	1554.564	2300	10371.3545	8071.3545
1100	5997.1157	4897.1157	3600	-200.925	3800.925	2000	-5296.2515	7296.2515
1500	-1400.2466	2900.2466	2900	2320.6804	579.3196	691	5345.856	4654.856
829	5109.559	4280.559	4400	-16981.311	21381.3105	1500	-8758.2705	10258.2705
cost : 861,177,000			cost : 341,634,000			cost : 224,911,000		

Linear Regression with normalization

training data set			validataion data set			test data set		
y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)
16100	3876.1433	12223.8567	2500	3746.094	1246.094	826	1654.2756	828.2756
508	1107.6248	599.6248	1300	3159.3393	1859.3393	920	2586.4453	1666.4453
1300	3500.4951	2200.4951	559	2570.6897	2011.6897	2300	2433.1396	133.1396
1100	1585.0162	485.0162	3600	3223.5186	376.4814	2000	3157.2802	1157.2802
1500	4043.2334	2543.2334	2900	2584.5952	315.4048	691	1527.3289	836.3289
829	1742.4027	913.4027	4400	4019.1997	380.8003	1500	5956.497	4456.497
cost : 68,357,600			cost : 32,991,000			cost : 84,110,500		

Polynomial Linear Regression with Regularization

training data set			validataion data set			test data set		
y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)
16100	3540.0029	12559.9971	2500	3631.0081	1131.0081	826	2691.9092	1865.9092
508	1035.3757	527.3757	1300	2809.5867	1509.5867	920	2184.0776	1264.0776
1300	3050.4766	1750.4766	559	2752.9633	2193.9633	2300	2852.9937	552.9937
1100	1593.5042	493.5042	3600	3064.5796	535.4204	2000	3029.8298	1029.8298
1500	4145.1685	2645.1685	2900	2732.2559	167.7441	691	1088.835	397.835
829	1736.196	907.196	4400	3873.854	526.146	1500	5907.6655	4407.6655
cost : 68,300,400			cost : 32,852,300			cost : 83,386,800		

Use only important theta values in Polynomial Linear Regression with Regularization

training data set			validataion data set			test data set		
y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)	y_data	y_data_hat	abs(y_data - y_data_hat)
16100	3734.5029	12365.4971	2500	3060.8228	560.8228	826	3411.1489	2585.1489
508	1325.782	817.782	1300	2497.6987	1197.6987	920	4311.3447	3391.3447
1300	3386.1663	2086.1663	559	2544.663	1985.663	2300	2443.1479	143.1479
1100	1897.0237	797.0237	3600	2811.8496	788.1504	2000	3483.4729	1483.4729
1500	3715.7927	2215.7927	2900	2768.9248	131.0752	691	2043.783	1352.783
829	2660.5144	1831.5144	4400	4003.8618	396.1382	1500	3811.9312	2311.9312
cost : 68,678,100			cost : 33,512,200			cost : 85,801,200		

실험 결과, cost 값이 낮을수록 예측 값과 실제 값 간의 차이도 비례하게 낮아짐을 알 수 있었다. 각 모델의 성능은 cost 를 기준으로 Polynomial Linear Regression with

Regularization, Linear Regression with normalization, Polynomial Linear Regression with Regularization and using important theta, Linear Regression without normalization 순으로 좋았다. 성능 개선 정도는 다음 표와 같다. 아래로 갈수록 성능이 개선된 모델이다.

	Training data set cost	Validation data set cost	Test data set
LR	-	-	-
Important theta	12.539 배	10.194 배	2.621 배
Normalization LR	1.004 배	1.015 배	1.020 배
Polynomial LR	1.001 배	1.004 배	1.009 배

단순 Linear Regression 과 나머지 3 개의 모델의 cost 차이는 상당하였지만, 나머지 3 개의 모델 간의 성능차이는 거의 없었다. 이를 미루어 보아, 우리가 사용한 data set 은 normalization 에 상당한 영향을 받지만, 우리가 실험적으로 진행한 Important theta 추출 방법, 단순화한 2 차 Polynomial Linear Regression 의 방법은 실험 결과에 크게 영향을 미치지 못함을 알 수 있었다.

3. 성과

1) normalization 으로 인한 성능 향상을 확인 할 수 있었다. 아마 이 사실은 최적의 Learning rate 의 값이 얼마나 개선 됐는지에 따라서 확인 할 수 있을 것이라 판단한다.

2) regularization 에서 λ 에 따라서 그래프가 펴진다는 사실을 실험을 통하여 예측 할 수 있었다.

3) 값이 큰 theta 가 실제 예측에 영향을 많이 미치지 않음을 알 수 있었다.

4. 개선 사항

1) Polynomial Linear Regression 에서 항의 개수에 따라서 cost 의 변화를 확인해 볼 필요가 있다.

2)

Debugging a Learning Algorithm

예를 들어, housing price를 예측하기 위해 regularized linear regression 을 사용한다고 하자. 그 때의 cost function은 다음과 같다.

$$J(\theta) = \frac{1}{2m}\left[\Sigma_{i=1}^{m}(h_\theta(x^{(i)})-y^{(i)})^2 + \lambda\Sigma_{j=1}^{m}\theta_j^2\right]$$

그런데 이 hypothesis를 새 데이터에 적용하자 예측 에러가 굉장히 컸다. 이 때 어떤 방법을 써서 성능을 개선할 수 있을까.

시도해볼 수 있는 방법들은 다음과 같다.

- Training example 을 늘린다.
- Feature 갯수를 줄인다.
- Polynomial feature를 추가한다.
- λ 를 늘린다 / 줄인다.

그러나 이 방법들을 마구잡이로 시도하기보다는 현재 알고리즘의 어느 부분에서 문제가 있는지 진단해서 체계적으로 알고리즘을 개선하는 편이 훨씬 효율적일 것이다.

Andrew Ng 의 6 주차 강의노트를 한글로 번역한 것을 찾아보면 cost 를 줄이는 방법에는 feature 의 개수를 줄여보는 방법이 있어서 cost 가 줄어들 것이라고 예측하였지만 아마 feature 의 theta 의 절대값이 1 이상인 것만 고른 방식이 실패 요인 일지도 모른다는 생각이 든다.