Predicting Pace Based on Previous Training Runs

Tiffany Jin
tjin1@stanford.edu
CS229: Machine Learning
December 12, 2014

1. Introduction

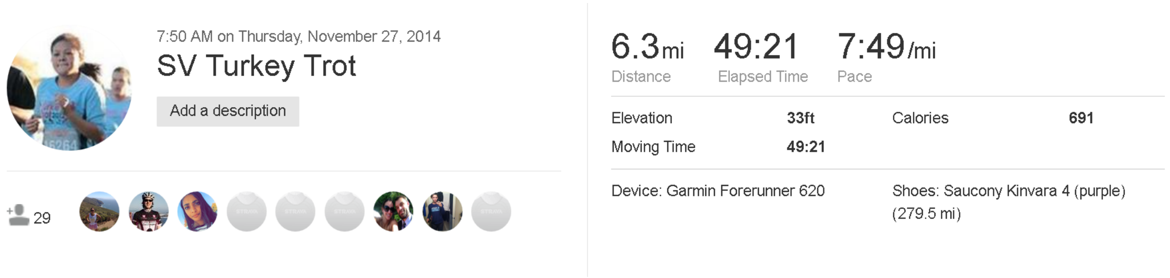
The era of social fitness has drawn an increasing number of people to the world of distance running. In the United States alone, 19 million people completed a running race in 2013, a 300 percent increase over those that did in 1990[1]. Furthermore, advances in GPS technology means that more and more people can carry their smart phones or a GPS-enabled watch in order to help track their run data. Instead of merely gauging average pace, runners can record their exact routes, as well as elevation changes and instantaneous pace. We can now definitively quantify progress in terms of new record mileage or seconds shaved off a familiar run segment. Additionally, online tracking and analysis sites such as Strava[2] have been able to document running history. Large sets of running data are being accumulated in both training and race situations. By studying the aggregated data from these websites, we can learn more about the nature of run training as well as how to best prepare for a race.

This project demonstrates how various machine learning algorithms can be applied to aggregate run data to predict a runner's pace for a specific route or segment. Using information about several different features from training runs, I will use several regression models to make predictions on pace during any given run. After I determine an effective model to predict the pace of any given training run, I will extrapolate this predicted pace to determine race pace by including a "race factor", which is a multiplier that accounts for faster pace on race day due to adrenaline.

2. Methods

Features:
For the purposes of creating a model to predict pace, I used three features: elevation gain, run mileage, and most recent 10k (6.2 miles) race time. Elevation gain and run mileage are raw data from Veloviewer[3], while most recent 10k time was obtained from the site Athlinks[4]. These three features are appropriate to the task of figuring out how fast a given person will run a given distance with a given elevation profile.

Data:
Strava is a mobile and online social fitness app that can be used to track routes, pace, elevation charts, and other relevant data during a run. A sample Strava activity is shown in Figure 1, with an example of some of the data that the site can record. The data on this site is stored in the form of .gpx, .tcx, and .fit files. Read-only access is provided to friends' data in the basic form seen in Figure 1.



7:50 AM on Thursday, November 27, 2014
SV Turkey Trot

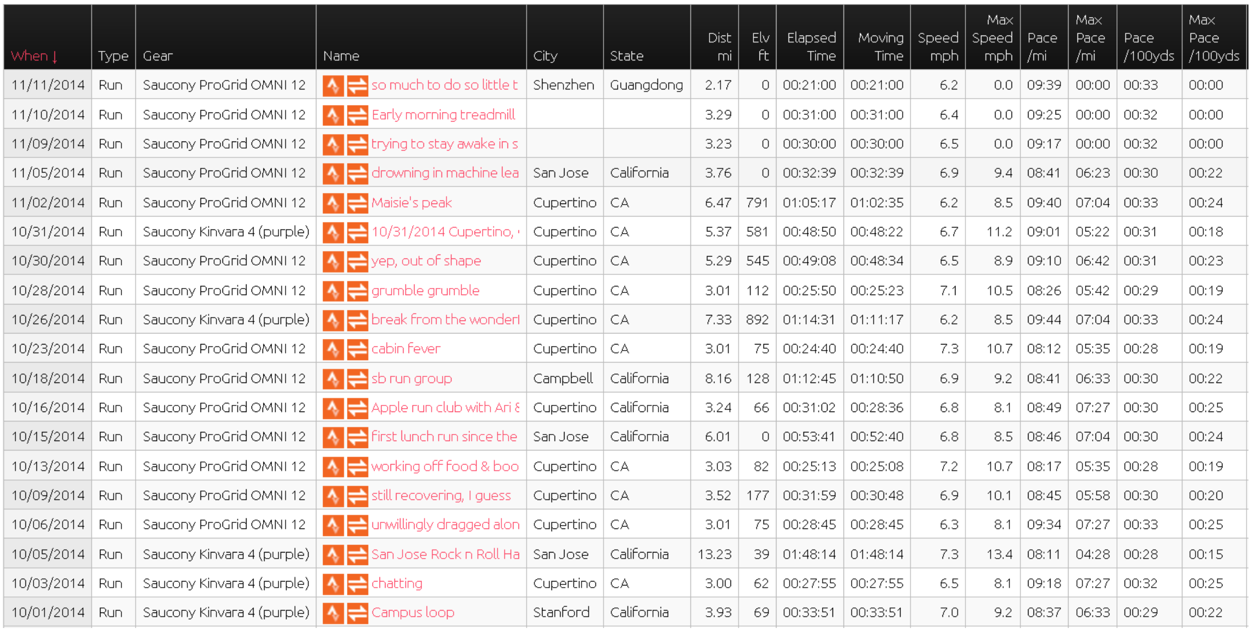
6.3mi	49:21	7:49/mi
Distance	Elapsed Time	Pace

Elevation	33ft	Calories	691
Moving Time	49:21		

Device: Garmin Forerunner 620
Shoes: Saucony Kinvara 4 (purple) (279.5 mi)

Figure 1: Example of Strava activity. Relevant data includes distance, moving time, pace, and elevation change.

Veloviewer is a website that compiles data from Strava and allows users to further analyze results beyond Strava's capabilities. Veloviewer uses the Strava API in order to do this. Due to privacy settings in the Strava API, a user can only analyze and export his or her own data. A partial sample of one user's data is provided in Figure 2. From this summary data, I was able to attain elevation gain and mileage for each training sample.



When ↓	Type	Gear	Name	City	State	Dist mi	Elv ft	Elapsed Time	Moving Time	Speed mph	Max Speed mph	Pace /mi	Max Pace /mi	Pace /100yds	Max Pace /100yds
11/11/2014	Run	Saucony ProGrid OMNI 12	so much to do so little t	Shenzhen	Guangdong	2.17	0	00:21:00	00:21:00	6.2	0.0	09:39	00:00	00:33	00:00
11/10/2014	Run	Saucony ProGrid OMNI 12	Early morning treadmill			3.29	0	00:31:00	00:31:00	6.4	0.0	09:25	00:00	00:32	00:00
11/09/2014	Run	Saucony ProGrid OMNI 12	trying to stay awake in s			3.23	0	00:30:00	00:30:00	6.5	0.0	09:17	00:00	00:32	00:00
11/05/2014	Run	Saucony ProGrid OMNI 12	drowning in machine lea	San Jose	California	3.76	0	00:32:39	00:32:39	6.9	9.4	08:41	06:23	00:30	00:22
11/02/2014	Run	Saucony ProGrid OMNI 12	Maisie's peak	Cupertino	CA	6.47	791	01:05:17	01:02:35	6.2	8.5	09:40	07:04	00:33	00:24
10/31/2014	Run	Saucony Kinvara 4 (purple)	10/31/2014 Cupertino, '	Cupertino	CA	5.37	581	00:48:50	00:48:22	6.7	11.2	09:01	05:22	00:31	00:18
10/30/2014	Run	Saucony ProGrid OMNI 12	yep, out of shape	Cupertino	CA	5.29	545	00:49:08	00:48:34	6.5	8.9	09:10	06:42	00:31	00:23
10/28/2014	Run	Saucony ProGrid OMNI 12	grumble grumble	Cupertino	CA	3.01	112	00:25:50	00:25:23	7.1	10.5	08:26	05:42	00:29	00:19
10/26/2014	Run	Saucony Kinvara 4 (purple)	break from the wonderl	Cupertino	CA	7.33	892	01:14:31	01:11:17	6.2	8.5	09:44	07:04	00:33	00:24
10/23/2014	Run	Saucony ProGrid OMNI 12	cabin fever	Cupertino	CA	3.01	75	00:24:40	00:24:40	7.3	10.7	08:12	05:35	00:28	00:19
10/18/2014	Run	Saucony ProGrid OMNI 12	sb run group	Campbell	California	8.16	128	01:12:45	01:10:50	6.9	9.2	08:41	06:33	00:30	00:22
10/16/2014	Run	Saucony ProGrid OMNI 12	Apple run club with Ari &	Cupertino	California	3.24	66	00:31:02	00:28:36	6.8	8.1	08:49	07:27	00:30	00:25
10/15/2014	Run	Saucony ProGrid OMNI 12	first lunch run since the	San Jose	California	6.01	0	00:53:41	00:52:40	6.8	8.5	08:46	07:04	00:30	00:24
10/13/2014	Run	Saucony ProGrid OMNI 12	working off food & boo	Cupertino	CA	3.03	82	00:25:13	00:25:08	7.2	10.7	08:17	05:35	00:28	00:19
10/09/2014	Run	Saucony ProGrid OMNI 12	still recovering, I guess	Cupertino	CA	3.52	177	00:31:59	00:30:48	6.9	10.1	08:45	05:58	00:30	00:20
10/06/2014	Run	Saucony ProGrid OMNI 12	unwillingly dragged alon	Cupertino	CA	3.01	75	00:28:45	00:28:45	6.3	8.1	09:34	07:27	00:33	00:25
10/05/2014	Run	Saucony Kinvara 4 (purple)	San Jose Rock n Roll Ha	San Jose	California	13.23	39	01:48:14	01:48:14	7.3	13.4	08:11	04:28	00:28	00:15
10/03/2014	Run	Saucony Kinvara 4 (purple)	chatting	Cupertino	CA	3.00	62	00:27:55	00:27:55	6.5	8.1	09:18	07:27	00:32	00:25
10/01/2014	Run	Saucony Kinvara 4 (purple)	Campus loop	Stanford	California	3.93	69	00:33:51	00:33:51	7.0	9.2	08:37	06:33	00:29	00:22

Figure 2: Example of Veloviewer data. Note that the same relevant data is available, but is now in a summary form.

Athlinks is another social fitness website geared at competitive endurance athletes. Its primary purpose is to record race results. Unlike Strava and Veloviewer, privacy settings are less restrictive and one can access anyone's race history given just their names. Using Athlinks, I was able to look up each runner's most recent 10k race time and add that to my feature set.

Using Matlab, I imported a total of 447 training examples from four different athletes. The data was randomly divided into five parts for use in 5-fold cross validation during the training of four different regression models.

3. Regression Models

Four regression models were used to predict a run pace. θ is a vector of regression coefficients, used to make a prediction on y given some parameters x (the feature set of elevation gain, mileage, and most recent 10k race).
General Regression Equation (for all models):

$$Y_{predicted} = \theta^T X_{test}$$

(1) Basic Linear Regression:
Basic linear regression is the simplest form of linear regression, and uses training examples x as well as labels y in order to attain regression coefficients θ.

$$\theta = (X^T X) X^T Y$$

(2) Locally Weighted Linear Regression:
Locally weighted linear regression more heavily weights training data that has similar features to the test data's features when making a prediction. $w^{(i)}$ indicates the weight of a particular training example, and τ indicates the bandwidth parameter.

$$\theta = (X^T W X) X^T Y$$

$$w^{(i)} = \exp\left(\frac{\left(x^{(i)} - x\right)^2}{2\tau^2}\right)$$

$$\tau = 10^6$$

(3) Ridge Regression:
Ridge regression and lasso regression are regularization techniques. When features are not independent, θ may be highly sensitive to small errors in y. Ridge regression incorporates small positive values of k to improve the conditioning of the problem and reduce the variance of the estimates.

$$\theta = (X^T X + kI)X^T Y$$
$$k = 0.01$$

(4) Lasso Regression:
Lasso regression, on the other hand, uses an L_1 instead of L_2 constraint. Unlike the previous models, there is no easy solution for θ, as it is nonlinear.

$$\text{Minimize: } \sum_{i=1}^{n}(Y_i - \sum_{j=1}^{m} X_{ij}\,\theta_j)^2 + \lambda \sum_{j=1}^{m}|\theta_j|$$
$$\lambda = 0.0098$$

4. Results

Each of the models was run using 5-fold cross validation, with training MSE (mean square error) the average of the MSE of those 5 training sets and test MSE the average of the MSE of those 5 test sets. After performing such cross validation on four linear regression-type models, I found that all of the models resulted in very similar generalization errors. Basic linear regression, as well as weighted, ridge regression, and lasso regression, showed little improvements to the resulting test MSE (mean square of errors), despite optimizing of parameters τ, k, and λ. Sample sizes and corresponding MSE to each model is shown in Table 1, and a bar graph comparing training and test MSE from each model is shown in Figure 3.

Model	Training samples	Test samples	Training MSE (error in minutes)	Test MSE (error in minutes)
Basic linear regression	358	89	0.6403	0.5517
Locally weighted regression	358	89	0.4624	0.5521
Ridge regression	358	89	0.4615	0.5559
Lasso regression	358	89	0.5722	0.5682

Table 1: Summary of results. Four-fifths of the total samples were training samples and one-fifth were test samples for each iteration during 5-fold cross validation. Note that results are given in terms of mean square error rather than percent error.

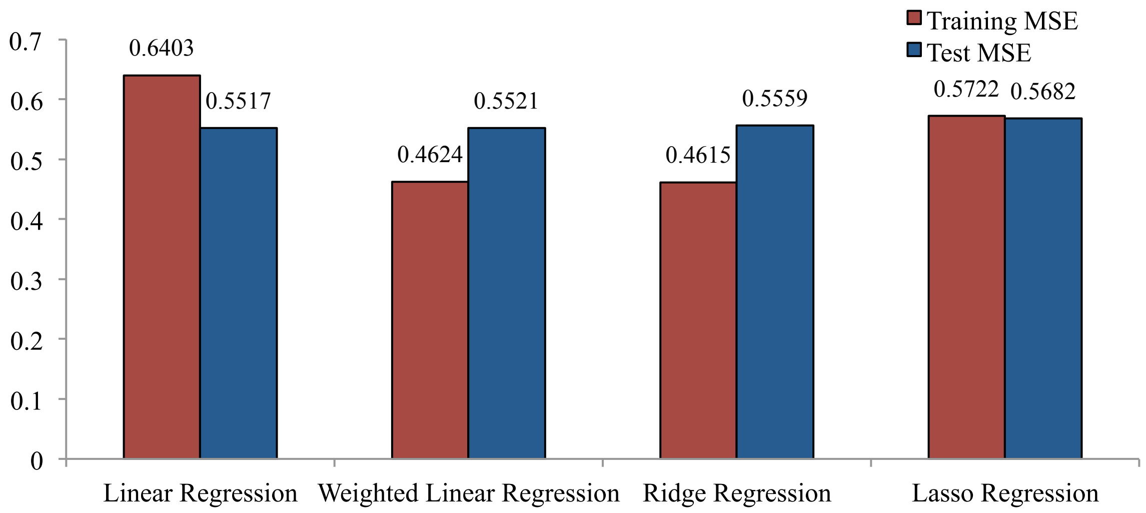


Figure 3: Bar graph showing MSE (mean square errors) of training and test sets for each model.

5. Discussion

Model limitations:
In comparison with simple linear regression, I expected weighted linear regression, ridge regression, and lasso regression to make reductions to the resulting test error. However, this was not the case as none of the models' test errors differed significantly even after optimization using model parameters. More specifically, insignificant improvements with locally weighted linear regression could be due to high variances in the data. Increasing the amount of training data available could perhaps reduce this variance and improve the usefulness of this regression model, but would also cause an increase in computational time. Another method of reducing the variance would be to include many more features, then do feature selection to identify the most relevant features.

Limited improvements with ridge regression can be explained because the three features used are not highly correlated. Ridge regression will improve a model when some features being used are not independent. Elevation gain and length of a run are somewhat correlated, as the longer a run the more opportunity for elevation gain. However, these features appear to be independent enough such that this form of regularization is not necessary and cannot limit the error any further.

Limited improvements with lasso regression can be explained because the regression model is highly dependent on only one of these features (most recent 10k). Lasso regression has a tendency to reduce less impactful features to zero, providing a built-in form of feature selection. While the hilliness of a run (elevation gain) and the length of a run both have some impact on the pace with which that route is completed, the largest contributor to this pace is a runner's most recent 10k time. As elevation gain and mileage were both already relatively insignificant features, using lasso regression did not much improve the model.

Finally, for all three models analysis could be limited by the linearity of these regression models. It is also possible that this problem is non-linear, and that using a non-linear regression model might be able to reduce the error further. It would be beneficial to try several other types of regression to get a better idea about model limitations, but I think it is more likely that the high variance seen in the training data was due to the limitations of the feature set.

Feature set limitations:
There were only three features used in training and evaluating this model. As stated previously, it is very likely that inclusion of several other features would be essential to creating a robust model and accurate predictor. The most important additional feature to be included would be heart rate. From the training data acquired, there was very high error due to large variations in effort during each run. This makes sense, since some days we are motivated, while on other days we are not. Many training plans often incorporate slower, lower-effort runs to prevent burnout, but the current model does not take training plans into consideration. The simplest way to quantify effort would be to record average heart rate for each run, normalizing for each athlete's typical resting heart rate and maximum heart rate. Currently, the best measure of heart rate is a heart rate monitor, which is most commonly a chest strap paired with a GPS watch. As this method is cumbersome and often uncomfortable, heart rate data is only a small percentage of all training runs. As technology continues to advance and built-in optical heart rate sensors become more accurate, I am confident that heart rate data will become ubiquitous.

Another feature that would be useful to incorporate is a classification of type of training run. An interval run, during which a runner sprints short segments and slowly jogs others, might have the same average pace as a steady recovery run. While the statistics might appear otherwise similar, the resulting fatigue will be different and may effect runs in the following few days. By including this classification, the model can include another feature to help quantify effort and better predict pace. This can be done through Strava's drop-down menu in an individual activity, or perhaps through an algorithm that can read an activity file and classify its data.

Lastly, another aspect of run training to account for is training frequency. Average miles run per week during the last month could be a useful feature to have. This makes sense, as athletes who work out more often are likely to improve and perform at close to his or her peak. In contrast, one who has not run or exercised in a while may be performing at only a fraction of his or her potential, as well as a fraction of his or her last 10k race pace. Likewise, the current model includes most recent 10k time, but does not account for how recent such a 10k occurred. Modifying this feature to be most recent 10k pace during the last month or alternatively adding a feature such as time since most recent 10k time would allow us to account for this. Since the raw data from Veloviewer already includes a timestamp, including these features would require only some data processing within Matlab.

Other considerations:
So far, these results were acquired from four different athletes, for a total of 447 training examples. Just as several more features may help to improve the model, having a much larger dataset could reduce variance and improve the robustness of the prediction. Similarly, in order to have more meaningful results, the training examples need to have greater variety and range across all features. For example, training examples for different athletes should include runs ranging from pancake flat routes to steep hill climbs. In addition, the data collected should include athletes with increased variety in recent 10k paces. Including the edge cases of both elite runners and beginner athletes would be helpful for model selection, in addition to revealing differences in the way different classes of runners approach and respond to training and performance situations.

6. Future Work

The next steps in refining this problem would be to define the model more clearly. As discussed, I would try several other methods of regression to try to better fit the data and see if that would reduce the error. Additionally, I would like to add more features, since I think this would reduce a lot of the variation observed in the training examples. I would like to add more training runs from a variety of different situations and athletes, from elite runners to those who do not have any significant running experience. After the model has been appropriately refined, I would like to collect race data and correlate race data with training data. This might be determined through a "race factor" that would help predict run pace during a race by calculating how much faster a particular athlete will run during a race compared to typical training.

7. References

(1) "2014 State of the Sport – Part III: Race Trends." Running USA. 14th July, 2014. Available: http://www.runningusa.org/2014-state-of-sport?returnTo=annual-reports. Accessed: 10th December, 2014.
(2) Strava [Online]. Available: http://www.strava.com. Accessed: 5th December, 2014.
(3) Lowe, Ben. *Veloviewer*[Online]. Available: http://www.veloviewer.com. Accessed: 5th December, 2014.
(4) Athlinks[Online]. Available: http://www.athlinks.com. Accessed: 5th December, 2014.