Assignment 8

Introduction:

Given a corpus of 5000+ movies from IMDB website, use R programming language to process the data and build a simple command line recommendation and predictionsystem.

The dataset contains 28 variables for 5043 movies, spanning across 100 years in 66 countries. There are 2399 unique director names, and thousands of actors/actresses. Below are the 28 variables:

"movie_title", "color", "num_critic_for_reviews", "movie_facebook_likes", "duration", "director_name", "director_facebook_likes", "actor_3_name", "actor_3_facebook_likes", "actor_2_name", "actor_2_facebook_likes", "actor_1_name", "actor_1_facebook_likes", "gross" "genres", "num_voted_users", "cast_total_facebook_likes", "facenumber_in_poster", "plot_keywords", "movie_imdb_link", "num_user_for_reviews", "language", "country", "content_rating", "budget" "title_year", "imdb_score", "aspect_ratio"

Objectives:

- Learn basics of R
- Learn to apply basic similarity measure

Tasks to be done:

- Recommend top 5 movies based on user interest (in the form movie name) (use Cosine Similarity and Euclidean Distance similarity measures).

Cosine Similarity:

$$CosSim(x, y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}} = \frac{\langle x, y \rangle}{||x||\ ||y||}$$

Here x_i's and y_i's are the features of the vectors of the two movies.

Euclidean Distance:

$$\mathrm{d}(\mathbf{p},\mathbf{q}) = \mathrm{d}(\mathbf{q},\mathbf{p}) = \sqrt{(q_1-p_1)^2 + (q_2-p_2)^2 + \cdots + (q_n-p_n)^2}$$

$$= \sqrt{\sum_{i=1}^{n}(q_i-p_i)^2}.$$

Here q_i's and p_i's are the features of the vectors of the two movies.

[When a query is input by the user, represent the query as a vector of features and find it's Euclidean distance and cosine similarity with all the other movie vectors, and output the top 5 movies excluding the query movie]

- Identify top 5 director actor duo (using jaccard similarity).

Jaccard similarity:

$$s_{AB} = \frac{|A \cap B|}{|A \cup B|}$$

Here A is a set of movies of the directors and B is a set of movies of Actors

NOTE: Exclude all the text features for simple calculations

- Prediction of IMDB score of a movie given its features using linear regression. (test data and training data will be provided)

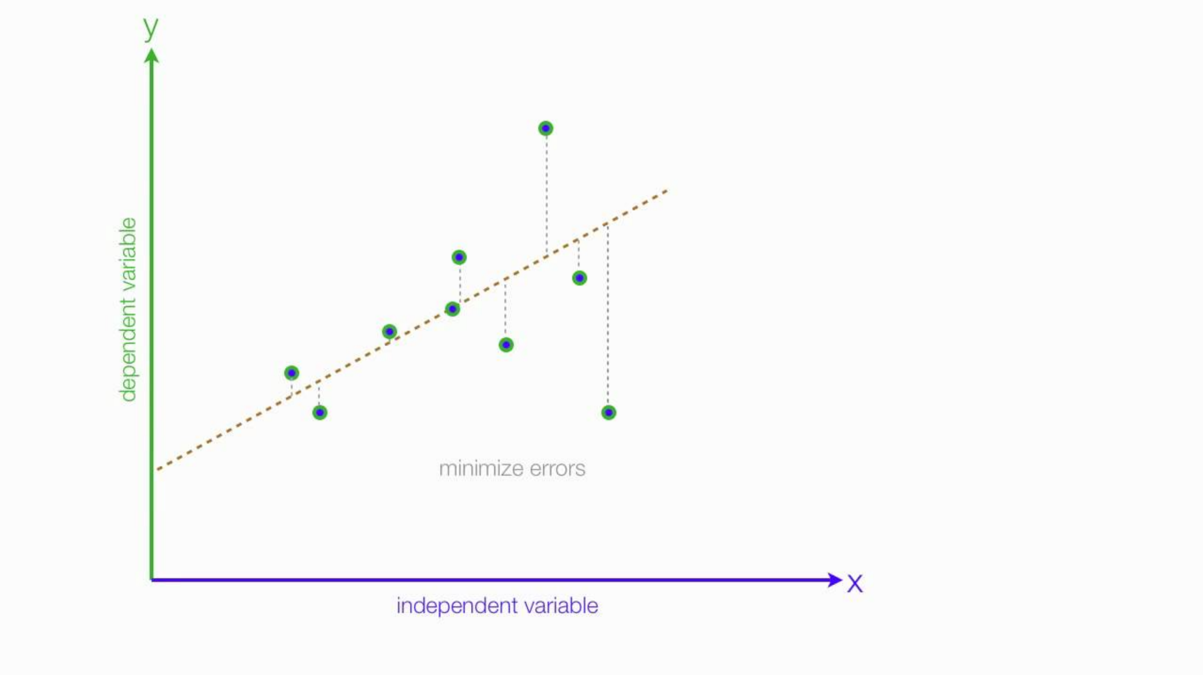
Linear Regression:

Linear regression analysis is the most widely used of all statistical techniques: it is the study of *linear*, *additive* relationships between variables. Let Y denote the "dependent" variable whose values you wish to predict, and let X_1, …,X_k denote the "independent" variables from which you wish to predict it, with the value of variable X_i in period t (or in row t of the data set) denoted by X_{it}. Then the equation for computing the predicted value of Y_t is:

$$\hat{\mathbf{Y}}_t = \mathbf{b}_0 + \mathbf{b}_1\mathbf{X}_{1t} + \mathbf{b}_2\mathbf{X}_{2t} + \ldots + \mathbf{b}_k\mathbf{X}_{kt}$$

This formula has the property that the prediction for Y is a straight-line function of each of the X variables, holding the others fixed, and the contributions of different X variables to the predictions are additive. The slopes of their individual straight-line relationships with Y are the constants $\mathbf{b_1}$, $\mathbf{b_2}$, …,$\mathbf{b_k}$, the so-called *coefficients* of the variables. That is, $\mathbf{b_i}$ is the change in the predicted value of Y per unit of change in X_i, other things being equal. The additional constant $\mathbf{b_0}$, the so-called *intercept*, is the prediction that the model would make if all theX's were zero (if that is possible).

The coefficients and intercept are estimated by *least squares*, i.e., setting them equal to the unique values that minimize the sum of squared errors within the sample of data to which the model is fitted. And the model's prediction errors are typically assumed to be *independently and identically normally distributed*.



- Plot the graph showing the original rating and the predicted rating of the test data.
- Evaluate your Linear Regression model using – RMSE (Root Mean Squared Error)

RMSE:
The square root of the mean/average of the square of all of the error.The use of RMSE is very common and it makes an excellent general purpose error metric for numerical predictions.Compared to the similar Mean Absolute Error, RMSE amplifies and severely punishes large errors.

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

Here y_i's are correct output variable and $\acute{y}_i$'s are predicted output.

- Make a test and train set and learn a classifier (K-Nearest Neighbor) to predict genre of a movie from the test set. Then calculate the prediction accuracy of your classifier.

Refer to the below link for the theoretical background of K-Nearest Neighbor.
http://10.5.30.126/moodle/mod/resource/view.php?id=2579
Prediction Accuracy = 1 – Error Rate
Error Rate can be expressed in terms of **MSE** (Mean Squared Error).

MSE:

If $\boldsymbol{Y}^{\wedge}$is a vector of n predictions, and ***Y***is the vector of observed values corresponding to the inputs to the function which generated the predictions, then the MSE of the predictor can be estimated by

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\hat{Y_i} - Y_i)^2$$