Marathon Match

Problem Statement

Problem: ChildStuntedness

Problem Statement

Stunting affects more than one in four children worldwide. Children with stunted growth have an increased risk of early death, higher burden of disease, compromised physical capacities, and diminished cognitive development. This can reduce the productivity of an entire generation.

The roots of stunted growth begin in the womb, leading to low birth weight infants entering the world at a deficit. Being able to predict, early in pregnancy, whether a child will have a low birth weight can help initiation of interventions leading to healthy live births. We need, then, to search not only for causes of low birth weight but also for methods for prediction of preterm birth outcomes (preterm babies are born before they spend the required 9 months in the womb).

Our goal is to determine a combination of early measures that would be a good predictor for birth weight. In pursuit of this goal, we have collected time series data from ultrasounds on pregnant mothers. We would like you to use this data to predict a child's birth weight and birth date (days from pregnancy start).

You may download the learning data set from here. The format for the data in the data set is a csv with details provided below:

Data Description

Column	Variable	Type	Label/Description
1	Id	int	Unique Fetus ID
2	t.ultsnd	float	Estimated fetus gestational age from last menstrual recall date
3	Sex	int	0 = Male, 1 = Female
4	Status	int	Maternal nutritional status (1 or 2)
5-12	Odv	float	Dependent variables: Ultrasound observed measurements
13	Birth Sz	float	Birth Weight (w)
14	Duration	float	Pregnancy Duration, or Birthday (b)

For each fetus given sex, status, and multiple ultrasound measurements(columns 5-12) during the pregnancy (time being the variable t.ultsnd). The data from the repeated ultrasounds provides a small time series that can be used for predicting the birth weight and day. More specifically each fetus has 6 ultrasounds done at regular intervals. For almost all IDs, the first ultrasound only one of 8 possible measurements is noted. For each remaining ultrasound

each of the remaining 7 measurements are noted almost every time (there are a few cases with missing values). An example of the measurements for a single fetus is shown below.

ID	t.ultsnd	Sex	Stat	1	2	3	4	5	6	7	8	w	b
1	0.221	0	2	0.049	NA	NA	NA	NA	NA	NA	NA	0.677	0.429
1	0.365	0	2	NA	0.304	0.260	0.321	0.253	0.264	0.244	0.015	0.677	0.429
1	0.525	0	2	NA	0.472	0.464	0.534	0.449	0.467	0.493	0.025	0.677	0.429
1	0.604	0	2	NA	0.592	0.528	0.618	0.523	0.517	0.592	0.030	0.677	0.429
1	0.812	0	2	NA	0.746	0.776	0.815	0.762	0.783	0.821	0.039	0.677	0.429
1	0.938	0	2	NA	0.799	0.875	0.880	0.854	0.888	0.873	0.042	0.677	0.429

(Note that NA values are reported as 0.000 in the CSV data.)

For each prediction (b_i, w_i), the error from the true birth date and birth weight will be measured as the squared Mahalanobis distance,

$$e_i = (b_i - b_{0i}, w_i - w_{0i})^T S^{-1} (b_i - b_{0i}, w_i - w_{0i})'$$

where S^{-1} is the inverse of the sample covariance matrix calculated on the complete dataset

where S-1 is the inverse of the sample covariance matrix calculated on the complete dataset.

```
inverseS[0][0] =  3554.42;  inverseS[0][1] = -328.119;
inverseS[1][0] =  -328.119; inverseS[1][1] =  133.511;
```

Scores will be calculated as a generalized R2 measure of fit. This is calculated as follows. The total sum of errors for the submission will be calculated as *SSE* = SUM(*ei*).

A baseline sum of squared error will be calculated by predicting the sample means for each fetus, that is the mean values of *b* and *w* for the current training set,

$$e_{0i} = (\bar{b} - b_{0i}, \bar{w} - w_{0i})S^{-1}(\bar{b} - b_{0i}, \bar{w} - w_{0i})'$$

SSE0 = SUM(*e0i*)

Then the submission score will be *Score* = 1000000 * MAX(1 - *SSE*/*SSE0*, 0).

In the String[] **trainingData**, each String states a record of some fetus, and has 14 tokens, comma-separated, in the same order as described above in the table. The format of **testingData** is almost same as the **trainingData**. The only difference is that the last two columns (the weight and the birth days) are missing. The datas with same IDs are consecutive. The returned double[] should contain the corresponding predictions for birthday (pregnancy duration) and weight of the fetus, for each ID, in numerical order by ID. More specifically, elements 0 and 1 represent the first fetus's birthday and weight, elements 2 and 3 the second fetus's birthday and weight, and so on. The length of the return array equals to the twice of the number of tested fetuses.

As an example, if the testing data contains several rows each for IDs 13, 4, and 9, then the return value should have six elements: {b4, w4, b9, w9, b13, w13}.

NOTE: All data values are normalized between 0 and 1 as part of data obfuscation requirements.

Notes on Data Set Generation

- The full data set contains approximately 28,000 lines, covering just over 4800 ID values.
- The full data set is divided into 20% for example tests, 30% for provisional tests, and 50% for system tests. All data belonging to the same ID is placed in the same data set.
- For each test, approximately 66% of the data (from that segment) is selected for training, and the remainder for testing.
- For provisional tests, all example data is also added to the training set.
- For system tests, all example and provisional test data is also added to the training set.

Definition

Class:	ChildStuntedness
Method:	predict
Parameters:	String[], String[]
Returns:	double[]
Method signature:	double[] predict(String[] training, String[] testing)
(be sure your method is public)	

Examples

0)

```
Seed: 1
```

1)

```
Seed: 2
```

2)

```
Seed: 3
```

3)

```
Seed: 4
```

4)

```
Seed: 5
```

5)

```
  Seed: 6
```
6)
```
  Seed: 7
```
7)
```
  Seed: 8
```
8)
```
  Seed: 9
```
9)
```
  Seed: 10
```
