

PERSONALIZED HEALTH INSURANCE PREDICTION USING ACTIVITY TRACKER



INTRODUCTION

THE IDEA



INTEGRATING DOMAINS



PROBLEM DEFINITION

By combining advanced analytics with wearable technology & personal information; insurers can give customers opportunities to take charge of their health insurance premium costs

DATASETS

Fitbit Web API

Healthcare Dataset

TECHNOLOGIES

Anaconda Navigator

-Numpy, Pandas, Seaborn, Matplotlib

APACHE SPARK 2.1.0

-Spark ML, SparkMLLib, SparkSQL

-pyspark

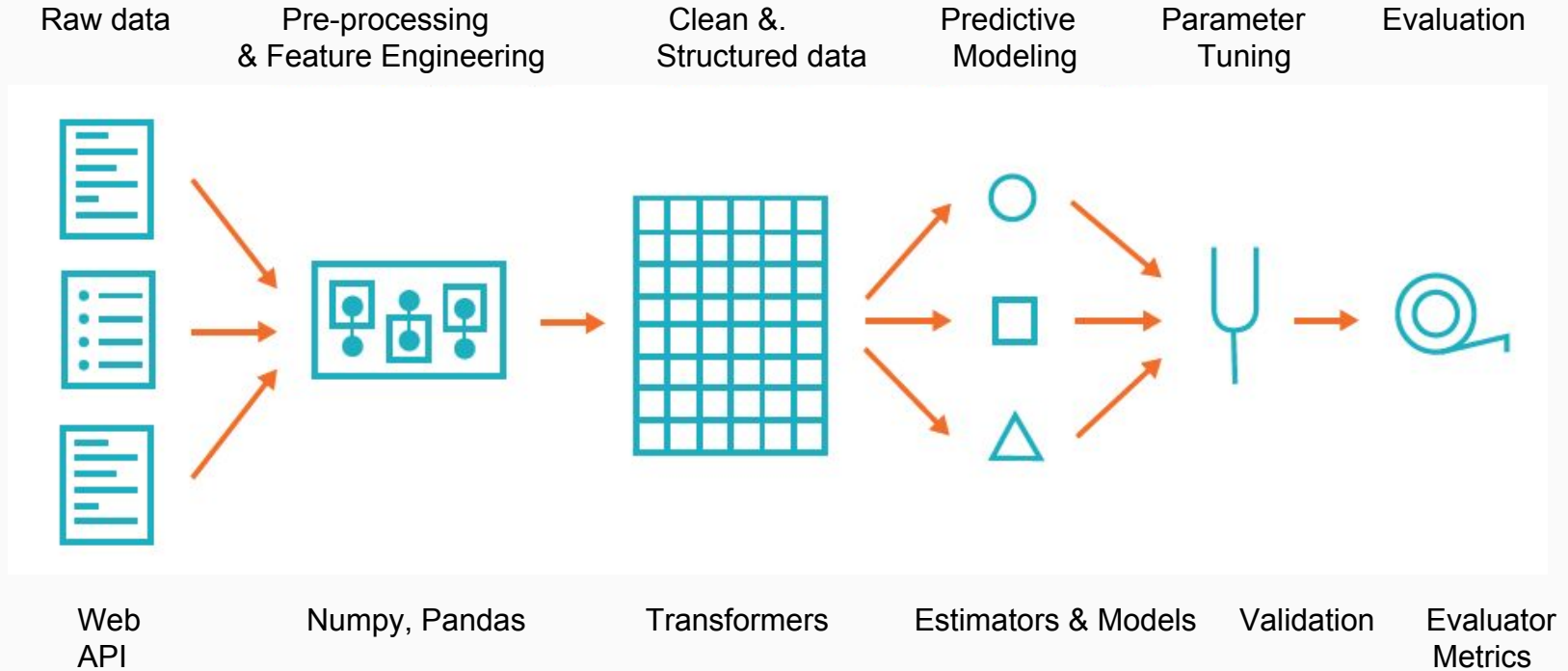
Docker

Zeppelin

Tableau



SYSTEM OVERVIEW



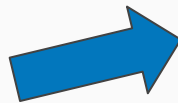
PROPOSED SYSTEM

EXTRACTION



HEALTHCARE DATASET

“age”, “sex”, “weight (in lbs)”, “BMI”,
“no of dependents”, “smoker?”



DataFrame

	Col1	Col2	Col3
Row 1				
Row 2				
Row 3				

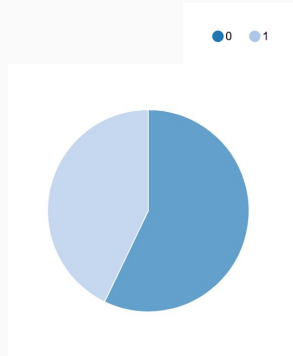


“heart-rate per minute”, “weight-logs”, “calories”,
“daily-activities”, “active-minutes{very active,
moderately active, fairly-active, sedentary-active}”

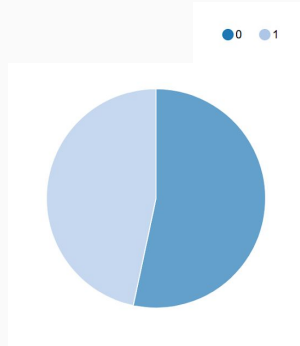


FEATURE IMPORTANCE

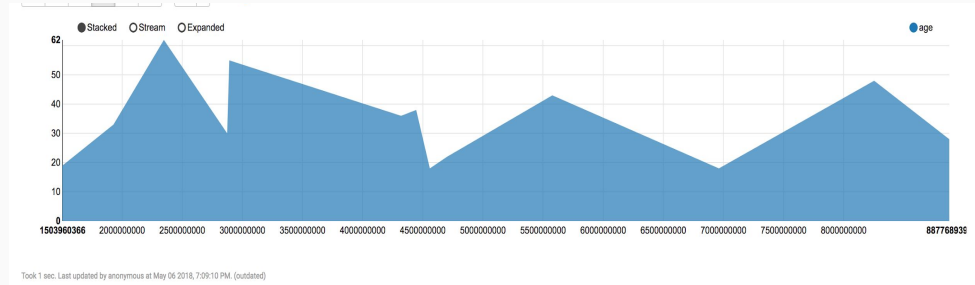
Sex wise Smoker Distribution



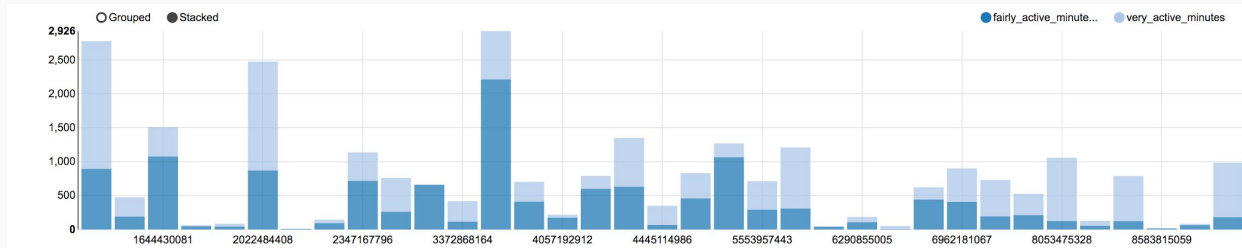
Sex wise Dependent Distribution



Age-range

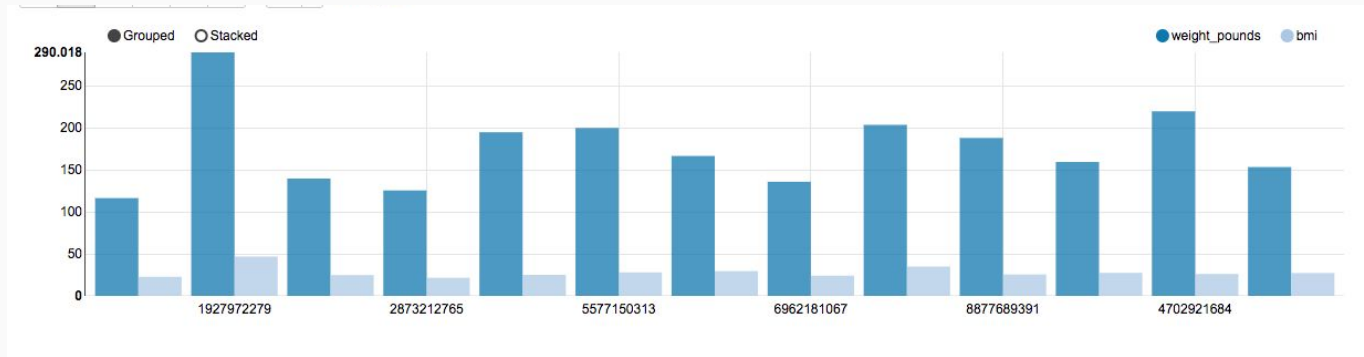


Mapping only fairly and very active minutes as they significantly contribute to the activity tracker and are relevant for insurance

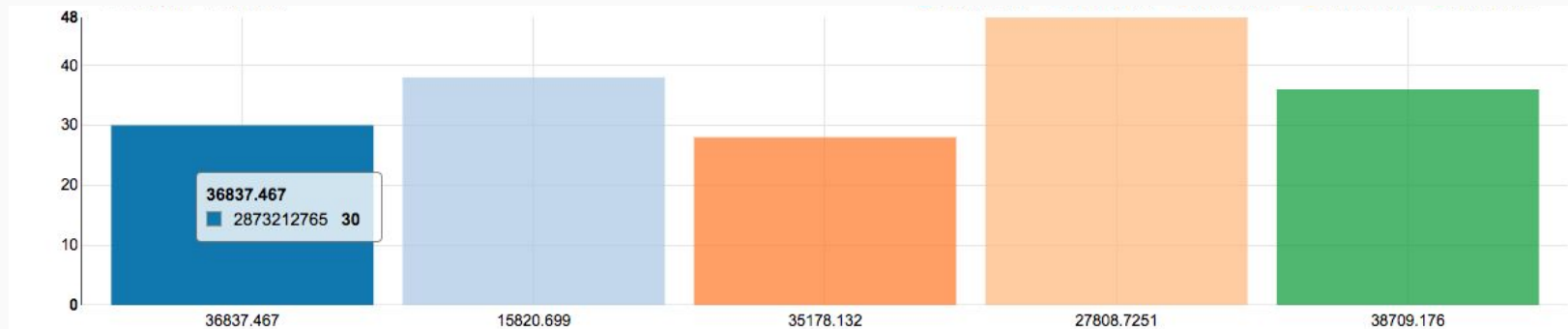


FEATURE IMPORTANCE

Weight & BMI for each unique participant



Health insurance cost mapped across age & smoker attribute



ANALYSIS: THE PHASES

Extract Fitbit Data
using Fitbit Web API

Handling missing values

Performed data manipulation by
converting categorical data

EXTRACTION

DATA PRE-PROCESSING

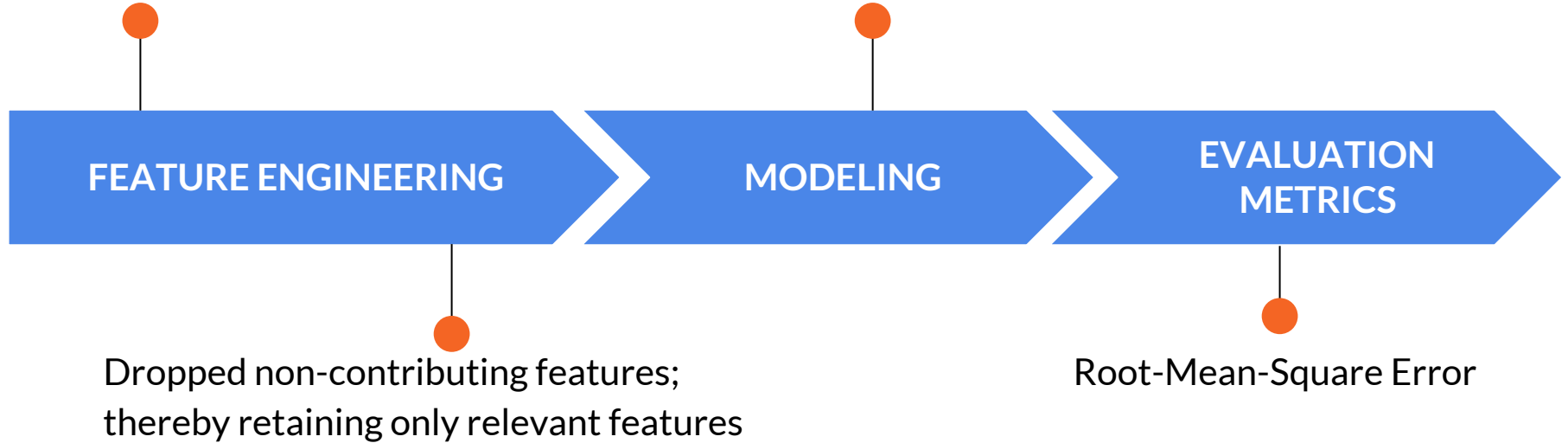
Extracting & Mapping data from
past health insurance records

Resolving inconsistencies
while coagulating data
from different sources

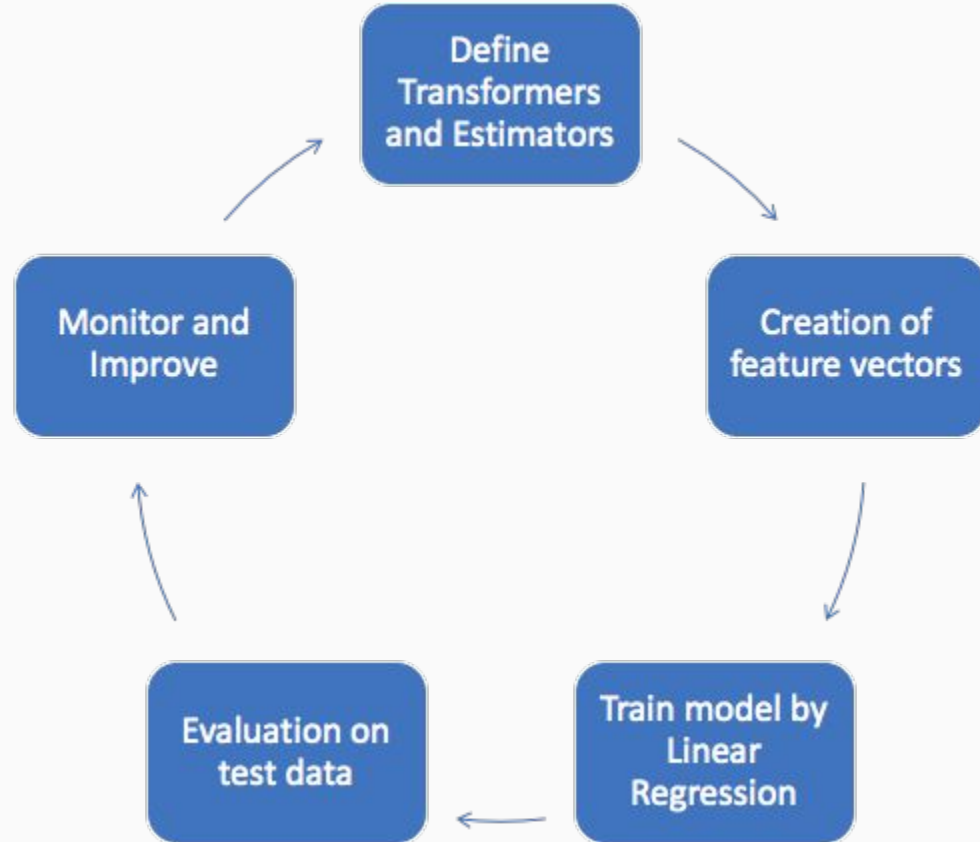
Renaming
column name
attributes

Fetches cleaned, structured & pre-processed data as an input

Performed Linear Regression by parameter tuning which best fits our model



PREDICTION & EVALUATION



INFORMATION VISUALIZATION

```
model: org.apache.spark.ml.feature.LinearRegressionModel = org.apache.spark.ml.feature.LinearRegressionModel$Builder$FromMLlib$.buildFromMLlib$(res661: Long = 31505, predictions: org.apache.spark.sql.DataFrame = [features: vector, label: int ... 1 more field])
Number of records accounted for prediction
res661: Long = 31505
predictions: org.apache.spark.sql.DataFrame = [features: vector, label: int ... 1 more field]
+-----+-----+-----+
|          features|label|          prediction|
+-----+-----+-----+
|[0.0,18.0,1.0,134...|21345|20936.050129858322|
```

RMSE as the evaluation metric to measure the accuracy

```
+-----+-----+-----+
only showing top 20 rows
rmse: Double = 324.8371506522351
324.8371506522351

Took 7 min 53 sec. Last updated by anonymous at May 06 2018, 5:49:17 PM. (outdated)
```

Helped the system to ingest and output larger data.

Scalable Linear Regression in large-scale environments.

CHALLENGES AND LIMITATIONS

Highly sensitive Data.

Data not readily available.

Use of consumer wearables within a clinical population is limited.

Customization of medical treatment.

Identify whether the drivers are likely to be involved in an accident, or have their car stolen.

Chronic Disease management.

THANK YOU.