In [1]:
```
import h2o
import pandas as pd
```

In [2]:
```
test_data = pd.read_csv('recruiting_zeta-disease_prediction-data_take-home-cha
llenge - 2021-01-21_zeta-disease_prediction-data_take-home-challenge.csv')
test_data.head()
```

Out[2]:

	age	weight	bmi	blood_pressure	insulin_test	liver_stress_test	cardio_stress_test	years_sm
0	24	151	39.5	69	72	1.3968	56	
1	27	179	35.5	89	156	1.6608	43	
2	34	147	26.9	76	74	1.6958	53	
3	35	206	32.4	73	127	1.4608	61	
4	60	193	29.8	62	192	1.7798	65	

In [3]:
```
#Check for issue that showed up in training data where years_smoking could be
 > age
#ideally will return 0 rows
check = test_data[test_data.years_smoking > test_data.age]
check.head(20)
```

Out[3]:

	age	weight	bmi	blood_pressure	insulin_test	liver_stress_test	cardio_stress_test	years_smo

In [4]:
```
h2o.init()
model = h2o.load_model('model')
```

```
Checking whether there is an H2O instance running at http://localhost:54321 .
connected.
```

H2O_cluster_uptime:	9 mins 19 secs
H2O_cluster_timezone:	America/New_York
H2O_data_parsing_timezone:	UTC
H2O_cluster_version:	3.36.0.3
H2O_cluster_version_age:	1 month and 7 days
H2O_cluster_name:	H2O_started_from_R_willi_zew062
H2O_cluster_total_nodes:	1
H2O_cluster_free_memory:	15.47 Gb
H2O_cluster_total_cores:	16
H2O_cluster_allowed_cores:	16
H2O_cluster_status:	locked, healthy
H2O_connection_url:	http://localhost:54321
H2O_connection_proxy:	{"http": null, "https": null}
H2O_internal_security:	False
Python_version:	3.6.13 final

In [5]:
```
#Read in replacements for 0's in bmi and blood_pressure that were determined d
uring model creation
replacements = pd.read_csv('replacement_vals.csv')
replacements.head()
```

Out[5]:

	blood_pressure	bmi
0	72.771053	32.682781

In [6]:
```
def cleanup_data(data, replacements):
    #Get rid of zeta_disease column for now
    data = data.drop('zeta_disease', axis = 1)

    #Replace cases where bmi = 0 with non-zero mean from training data
    #Even if 0's aren't showing up now, the goal is to replicate production lo
gic
    #where new data could have the same problem
    data.loc[data.bmi <= 0, 'bmi'] = replacements.bmi[0]

    #Replace cases where blood_pressure = 0 with non-zero mean from training d
ata
    data.loc[data.blood_pressure <= 0, 'blood_pressure'] = replacements.blood_
pressure[0]
    return(data)
```

In [7]:
```
model_input = cleanup_data(test_data, replacements)
```

In [8]:
```
model_input.head()
```

Out[8]:

	age	weight	bmi	blood_pressure	insulin_test	liver_stress_test	cardio_stress_test	years_sm
0	24	151	39.5	69.0	72	1.3968	56	
1	27	179	35.5	89.0	156	1.6608	43	
2	34	147	26.9	76.0	74	1.6958	53	
3	35	206	32.4	73.0	127	1.4608	61	
4	60	193	29.8	62.0	192	1.7798	65	

In [9]:
```
def create_predictions(data, model):
    #Convert data to h2o object
    data_h20 = h2o.H2OFrame(data)

    #Use model created in R to create predictions on new data
    predictions = model.predict(data_h20)

    #Append prediction back on to original dataframe
    predictions_pd = predictions.as_data_frame()

    output_dat = data
    output_dat['zeta_disease'] = predictions_pd.predict
    return(output_dat)
```

In [10]:
```
final_predictions = create_predictions(model_input, model)
```

```
Parse progress: |████████████████████████████████████████████████████
████| (done) 100%
drf prediction progress: |██████████████████████████████████████████████
████| (done) 100%
```

In [11]:
```
final_predictions.head(20)
```

Out[11]:

	age	weight	bmi	blood_pressure	insulin_test	liver_stress_test	cardio_stress_test	years_s
0	24	151	39.5	69.0	72	1.3968	56	
1	27	179	35.5	89.0	156	1.6608	43	
2	34	147	26.9	76.0	74	1.6958	53	
3	35	206	32.4	73.0	127	1.4608	61	
4	60	193	29.8	62.0	192	1.7798	65	
5	45	120	36.5	108.0	50	1.2978	54	
6	20	139	38.2	61.0	77	1.5818	68	
7	23	137	31.2	70.0	73	1.4168	59	
8	36	195	30.5	59.0	141	1.4498	59	
9	19	193	25.8	84.0	66	1.7938	50	
10	47	216	34.7	70.0	170	1.7238	58	
11	40	200	30.4	69.0	128	1.3118	60	
12	21	154	46.5	88.0	121	1.2498	68	
13	52	196	31.3	90.0	167	1.9238	66	
14	30	181	37.4	93.0	157	2.0508	80	
15	46	213	26.5	70.0	133	1.4788	55	
16	29	173	50.7	91.0	221	1.4878	83	
17	36	202	42.8	72.0	273	1.8748	72	
18	27	197	29.1	72.0	362	1.4298	69	
19	44	184	33.9	104.0	141	1.3268	60	

In [12]:
```
final_predictions.to_csv('PROJECT DELIVERABLES/results.csv', index = False)
```

In [ ]:
```

```