Analysing anomaly in king-country house process data

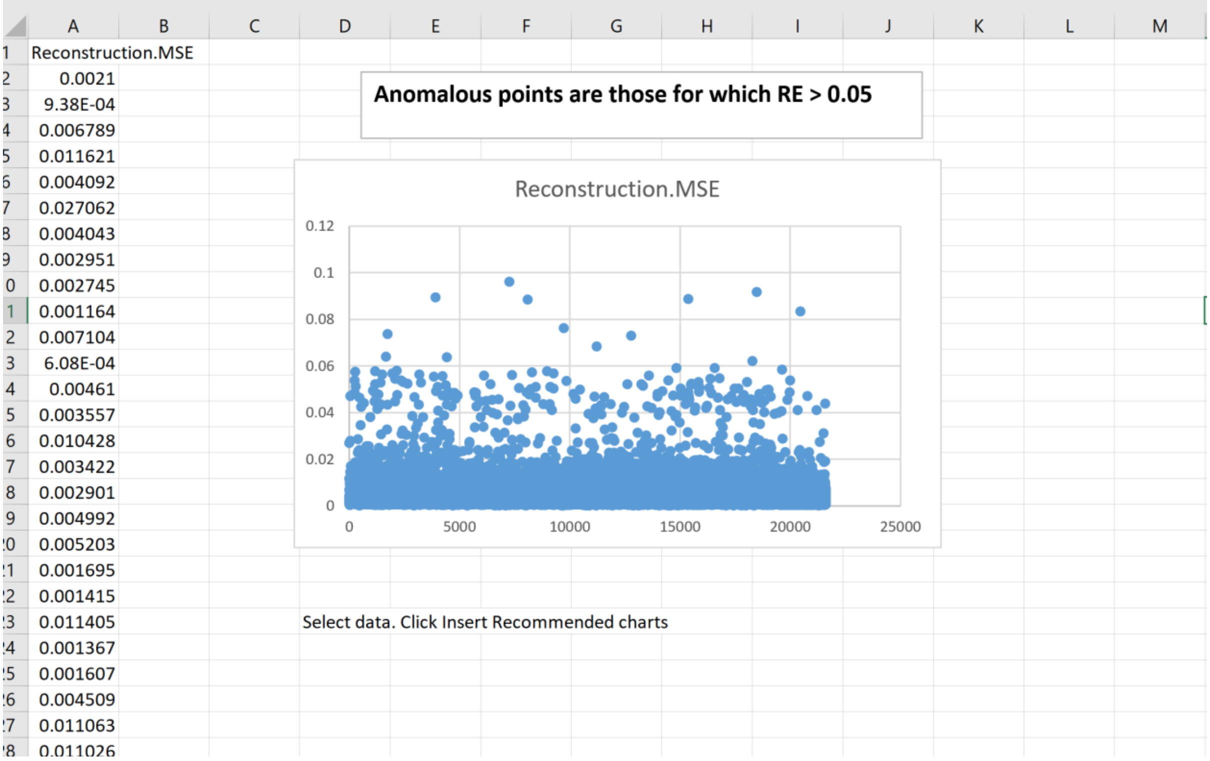
#Last amended: 31st Dec, 2023
#My folder: C:\Users\Ashok\OneDrive\Documents\king_country
# Kaggle: https://www.kaggle.com/datasets/harlfoxem/housesalesprediction

# Ref: h2o:

a. https://github.com/h2oai/h2o-tutorials/tree/master/best-practices/anomaly-detection
b. https://github.com/h2oai/h2o-tutorials/tree/master/best-practices

Steps:

a. Import data in h2o.ai flow
b. Do not split frame
c. Ignore *id* and *date* features
d. Build autoencoder model as given at the end of this document.
e. Predict Reconstruction error of original data.
f. Download predictions of MSE after combining with original data frame
g. Take the csv file to Windows 10 and open it in Excel
h. Perform plotting



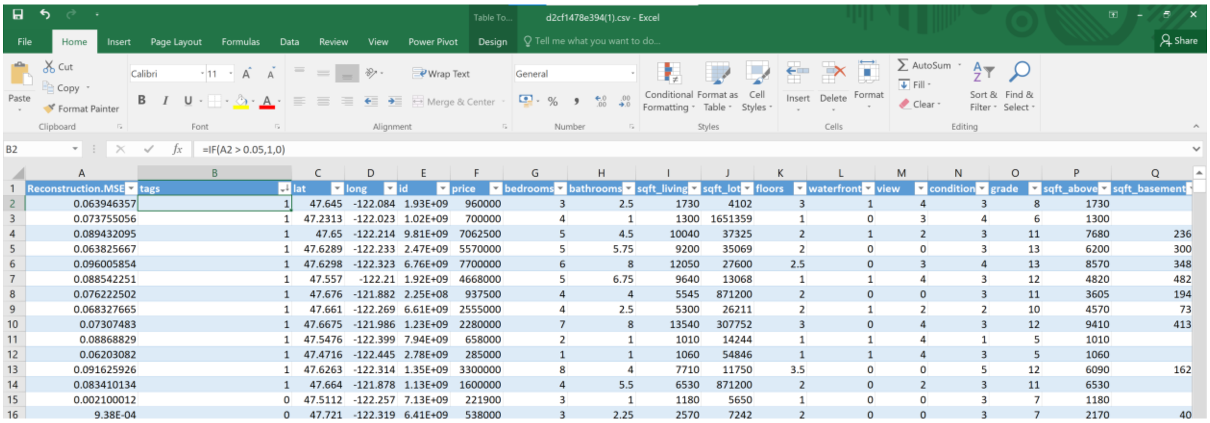
Reconstruction.MSE
0.0021
9.38E-04
0.006789
0.011621
0.004092
0.027062
0.004043
0.002951
0.002745
0.001164
0.007104
6.08E-04
0.00461
0.003557
0.010428
0.003422
0.002901
0.004992
0.005203
0.001695
0.001415
0.011405
0.001367
0.001607
0.004509
0.011063
0.011026

Anomalous points are those for which RE > 0.05

Select data. Click Insert Recommended charts

Figure 1: Reconstruction error plot

Excel sheet with reconstruction error (RE) and other features. Column '*tags*' has been derived from RE. RE > 0.06 is 1 else 0.



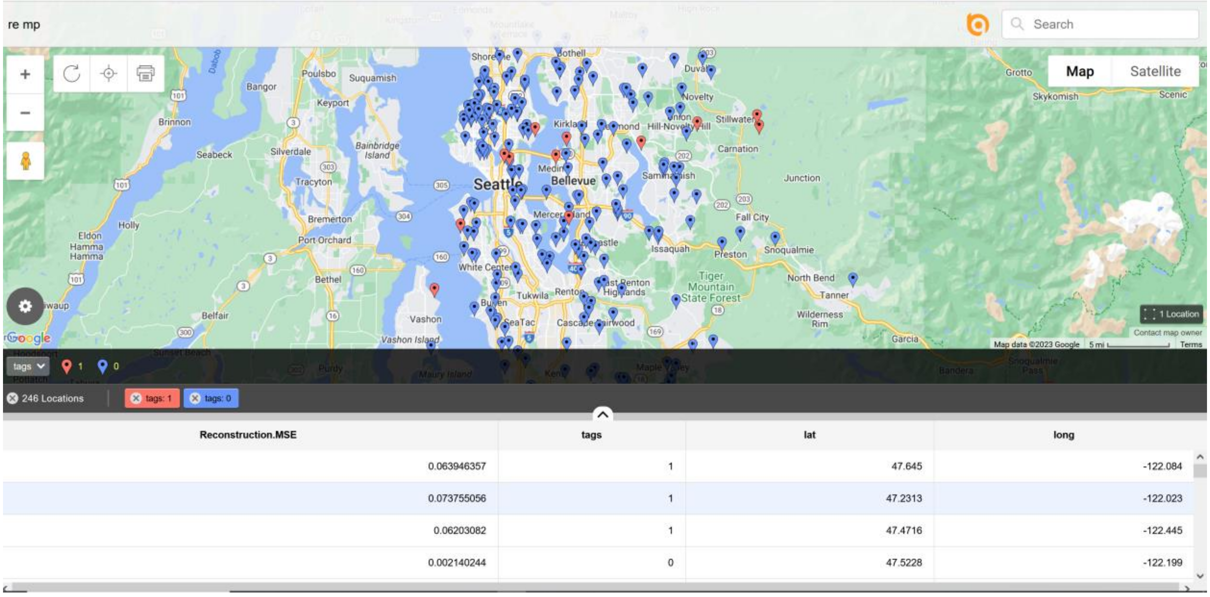
B2 =IF(A2 > 0.05,1,0)

	Reconstruction.MSE	tags	lat	long	id	price	bedrooms	bathrooms	sqft_living	sqft_lot	floors	waterfront	view	condition	grade	sqft_above	sqft_basement
2	0.063946357	1	47.645	-122.084	1.93E+09	960000	3	2.5	1730	4102	3	1	4	3	8	1730	
3	0.073755056	1	47.2313	-122.023	1.02E+09	700000	4	1	1300	1651359	1	0	3	4	6	1300	
4	0.089432095	1	47.65	-122.214	9.81E+09	7062500	5	4.5	10040	37325	2	1	2	3	11	7680	236
5	0.063825667	1	47.6289	-122.233	2.47E+09	5570000	5	5.75	9200	35069	2	0	0	3	13	6200	300
6	0.096005854	1	47.6298	-122.323	6.76E+09	7700000	6	8	12050	27600	2.5	0	3	4	13	8570	348
7	0.088542251	1	47.557	-122.21	1.92E+09	4668000	5	6.75	9640	13068	1	1	4	3	12	4820	482
8	0.076222502	1	47.676	-121.882	2.25E+08	937500	4	4	5545	871200	2	0	0	3	11	3605	194
9	0.068327665	1	47.661	-122.269	6.61E+09	2555000	4	2.5	5300	26211	2	1	2	2	10	4570	73
10	0.07307483	1	47.6675	-121.986	1.23E+09	2280000	7	8	13540	307752	3	0	4	3	12	9410	413
11	0.08868829	1	47.5476	-122.399	7.94E+09	658000	2	1	1010	14244	1	1	4	1	5	1010	
12	0.06203082	1	47.4716	-122.445	2.78E+09	285000	1	1	1060	54846	1	1	4	3	5	1060	
13	0.091625926	1	47.6263	-122.314	1.35E+09	3300000	8	4	7710	11750	3.5	0	0	5	12	6090	162
14	0.083410134	1	47.664	-121.878	1.13E+09	1600000	4	5.5	6530	871200	2	0	2	3	11	6530	
15	0.002100012	0	47.5112	-122.257	7.13E+09	221900	3	1	1180	5650	1	0	0	3	7	1180	
16	9.38E-04	0	47.721	-122.319	6.41E+09	538000	3	2.25	2570	7242	2	0	0	3	7	2170	40

Figure 2: Note the IF condition in the IInd column. The above table is sorted by tag values.

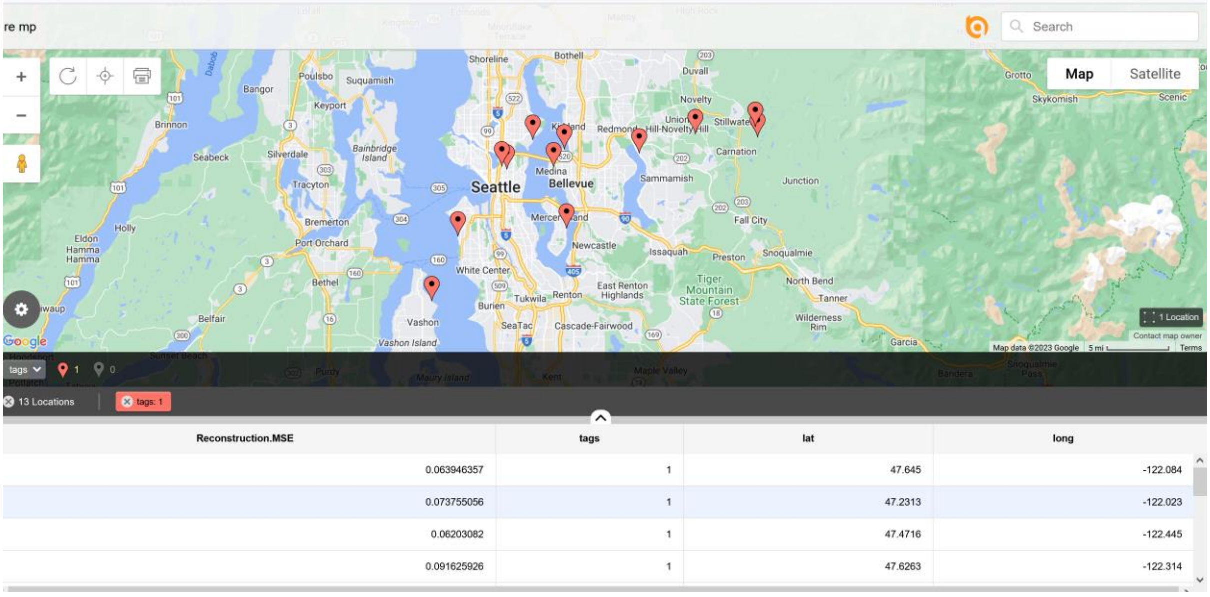
Sort the Excel sheet by *tags* and select top 250 points for plotting in batchgeo.com. Free version of batchgeo.com can take at most 250 points as a csv file—just drag and drop csv file to plot.

Map displaying high Reconstruction Error (points are in red):



Reconstruction.MSE	tags	lat	long
0.063946357	1	47.645	-122.084
0.073755056	1	47.2313	-122.023
0.06203082	1	47.4716	-122.445
0.002140244	0	47.5228	-122.199

Figure 3: tags 1 and tags 0 both getting displayed.



Reconstruction.MSE	tags	lat	long
0.063946357	1	47.645	-122.084
0.073755056	1	47.2313	-122.023
0.06203082	1	47.4716	-122.445
0.091625926	1	47.6263	-122.314

Figure 4 Only tag 1 getting displayed

Most of the tag 1 points are near waterfront.

```
# H2o: king-country autoencoder model

buildModel 'deeplearning', {
                              "model_id":"deeplearning-autoencoder",
                              "training_frame":"kc_house_data.hex", <== Full frame, no split
                              "nfolds":0,
                              "ignored_columns":["id","date"],      <==
                              "ignore_const_cols":true,
                              "activation":"Tanh",                  <==
                              "hidden":[5],                         <==
                              "epochs":100,                         <==
                              "variable_importances":false,         <==
                              "score_each_iteration":false,
                              "standardize":true,
                              "train_samples_per_iteration":-2,
                              "adaptive_rate":true,
                              "input_dropout_ratio":0,
                              "l1":0,
                              "l2":0,
                              "loss":"Automatic",
                              "distribution":"AUTO",
                              "quantile_alpha":0.5,
                              "huber_alpha":0.9,
                              "score_interval":5,
                              "score_training_samples":10000,
                              "score_validation_samples":0,
                              "score_duty_cycle":0.1,
                              "stopping_rounds":5,
                              "stopping_metric":"AUTO",
                              "stopping_tolerance":0,
                              "max_runtime_secs":0,
                              "autoencoder":true,
                              "categorical_encoding":"AUTO",
                              "auc_type":"AUTO",
                              "gainslift_bins":-1,
                              "overwrite_with_best_model":true,
                              "target_ratio_comm_to_comp":0.05,
                              "seed":-1,
                              "rho":0.99,
                              "epsilon":1e-8,
                              "nesterov_accelerated_gradient":true,
                              "max_w2":3.4028235e+38,
                              "initial_weight_distribution":"UniformAdaptive",
                              "classification_stop":0,
                              "regression_stop":0.000001,
                              "score_validation_sampling":"Uniform",
        "diagnostics":true,
        "fast_mode":true,
        "force_load_balance":true,
        "single_node_mode":false,
        "shuffle_training_data":true,                <==
        "missing_values_handling":"MeanImputation",
        "quiet_mode":false,
        "sparse":false,
        "col_major":false,
        "average_activation":0,
        "sparsity_beta":0,
        "max_categorical_features":2147483647,
        "reproducible":false,
        "export_weights_and_biases":false,
        "mini_batch_size":1,
        "elastic_averaging":false
}
```

#############