# eda

September 16, 2019

[1]:
```
# competition src: https://www.kaggle.com/c/forest-cover-type-prediction
# data src: https://www.kaggle.com/c/3936/download-all
```

[2]:
```
import gc
gc.collect()
```

[2]:
```
82
```

[160]:
```
# Load libraries

import re

import numpy as np

%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns

import pandas as pd
```

[4]:
```
# Load train and test dataset

TRAIN_FILEPATH = 'data/train.csv'
train_df = pd.read_csv( TRAIN_FILEPATH, header=0 )

TEST_FILEPATH = 'data/test.csv'
test_df = pd.read_csv( TEST_FILEPATH, header=0 )
```

[5]:
```
def overview_df( df ):
    display( df.sample() )
    display( df.shape )
    display( df.isnull().sum() )
    display( df.duplicated().sum() )
    df.info()
```

[96]:
```
overview_df( train_df )

overview_df( test_df )
```

```
     Elevation  Aspect  Slope  Horizontal_Distance_To_Hydrology  \
```

```
2101       2707     102     30                                150

      Vertical_Distance_To_Hydrology  Horizontal_Distance_To_Roadways  \
2101                              76                              150

      Hillshade_9am  Hillshade_Noon  Hillshade_3pm  \
2101            253             187             40

      Horizontal_Distance_To_Fire_Points  Wilderness_Area  Soil_Type  \
2101                                 765                1         30

      Cover_Type
2101           5


(15120, 13)


Elevation                           0
Aspect                              0
Slope                               0
Horizontal_Distance_To_Hydrology    0
Vertical_Distance_To_Hydrology      0
Horizontal_Distance_To_Roadways     0
Hillshade_9am                       0
Hillshade_Noon                      0
Hillshade_3pm                       0
Horizontal_Distance_To_Fire_Points  0
Wilderness_Area                     0
Soil_Type                           0
Cover_Type                          0
dtype: int64


0


<class 'pandas.core.frame.DataFrame'>
RangeIndex: 15120 entries, 0 to 15119
Data columns (total 13 columns):
Elevation                           15120 non-null int64
Aspect                              15120 non-null int64
Slope                               15120 non-null int64
Horizontal_Distance_To_Hydrology    15120 non-null int64
Vertical_Distance_To_Hydrology      15120 non-null int64
Horizontal_Distance_To_Roadways     15120 non-null int64
Hillshade_9am                       15120 non-null int64
Hillshade_Noon                      15120 non-null int64
Hillshade_3pm                       15120 non-null int64
```

```
Horizontal_Distance_To_Fire_Points    15120 non-null int64
Wilderness_Area                       15120 non-null int64
Soil_Type                             15120 non-null int64
Cover_Type                            15120 non-null int64
dtypes: int64(13)
memory usage: 1.5 MB
```

```
        Elevation  Aspect  Slope  Horizontal_Distance_To_Hydrology  \
109550       2902      41      9                               242

        Vertical_Distance_To_Hydrology  Horizontal_Distance_To_Roadways  \
109550                              35                             2885

        Hillshade_9am  Hillshade_Noon  Hillshade_3pm  \
109550            221             220            133

        Horizontal_Distance_To_Fire_Points  Wilderness_Area  Soil_Type
109550                                2233                1         29
```

```
(565892, 12)
```

```
Elevation                           0
Aspect                              0
Slope                               0
Horizontal_Distance_To_Hydrology    0
Vertical_Distance_To_Hydrology      0
Horizontal_Distance_To_Roadways     0
Hillshade_9am                       0
Hillshade_Noon                      0
Hillshade_3pm                       0
Horizontal_Distance_To_Fire_Points  0
Wilderness_Area                     0
Soil_Type                           0
dtype: int64
```

```
0
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 565892 entries, 0 to 565891
Data columns (total 12 columns):
Elevation                           565892 non-null int64
Aspect                              565892 non-null int64
Slope                               565892 non-null int64
Horizontal_Distance_To_Hydrology    565892 non-null int64
Vertical_Distance_To_Hydrology      565892 non-null int64
```

```
Horizontal_Distance_To_Roadways        565892 non-null int64
Hillshade_9am                          565892 non-null int64
Hillshade_Noon                         565892 non-null int64
Hillshade_3pm                          565892 non-null int64
Horizontal_Distance_To_Fire_Points     565892 non-null int64
Wilderness_Area                        565892 non-null int64
Soil_Type                              565892 non-null int64
dtypes: int64(12)
memory usage: 51.8 MB
```

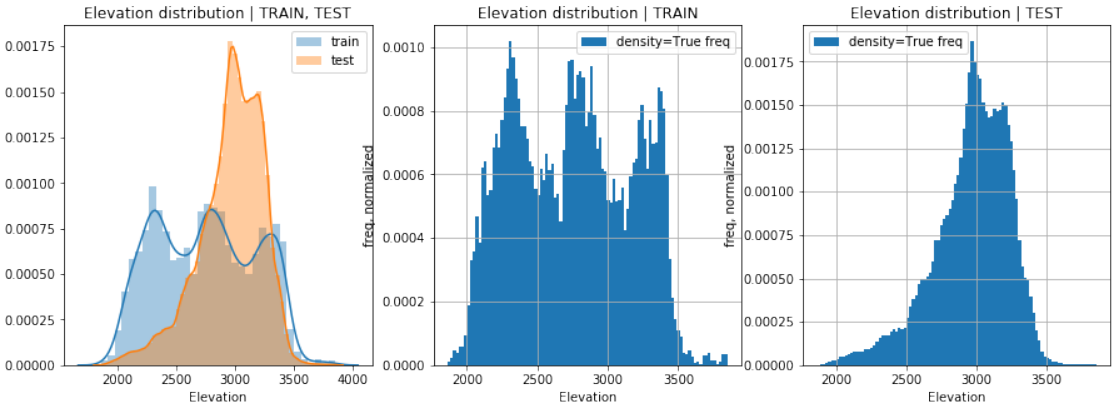
[7]:
```
# Transform multiple Wilderness_Areax or Soil_Typex-like features a into single␣
 ↪categorical feature

def merge_onehot( dataset_df, col_name_no_x ):
    """Convert col_nameX features to single feature
    X means some integer value.
    Doesn't work with multiple calls - returns 0s for all col_name_no_x.
    """
    dataset_df_cpy = dataset_df.copy()
    # 1. Identify columns
    all_df_columns = dataset_df_cpy.columns.values
    re_pattern_compiled = re.compile( "^{0}(\d+)$".format( col_name_no_x ) )
    matched_columns = list(filter( re_pattern_compiled.match, all_df_columns ))
    # 2. Change columns: multiply by 'x' value
    for matched_column in matched_columns:
        col_name_x_value = re_pattern_compiled.match(matched_column).groups()[0]
        dataset_df_cpy[matched_column] *= int( col_name_x_value )
    # 3. Merge col_namex columns into single col_name column
    dataset_df_cpy[col_name_no_x] = 0
    for matched_column in matched_columns:
        dataset_df_cpy[col_name_no_x] += dataset_df_cpy[matched_column]
    # 4. Drop col_namex columns
    dataset_df_cpy = dataset_df_cpy.drop( matched_columns, axis=1 )

    return dataset_df_cpy

def _ugly_merge_wildernessarea_soiltype_traintest( train_or_test_df ):
    train_or_test_df = merge_onehot( train_or_test_df,␣
 ↪col_name_no_x='Wilderness_Area' )
    train_or_test_df = merge_onehot( train_or_test_df,␣
 ↪col_name_no_x='Soil_Type' )
    return train_or_test_df
```

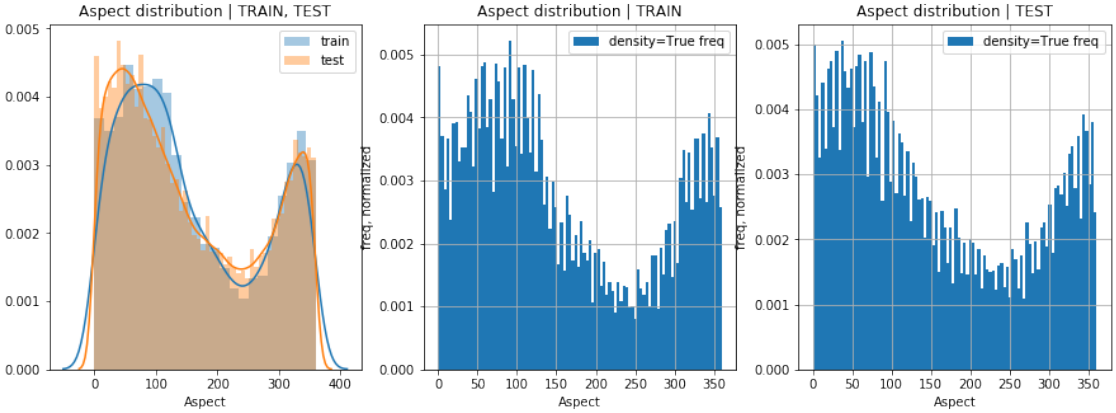
[8]:
```
train_df = _ugly_merge_wildernessarea_soiltype_traintest( train_df )

test_df = _ugly_merge_wildernessarea_soiltype_traintest( test_df )
```

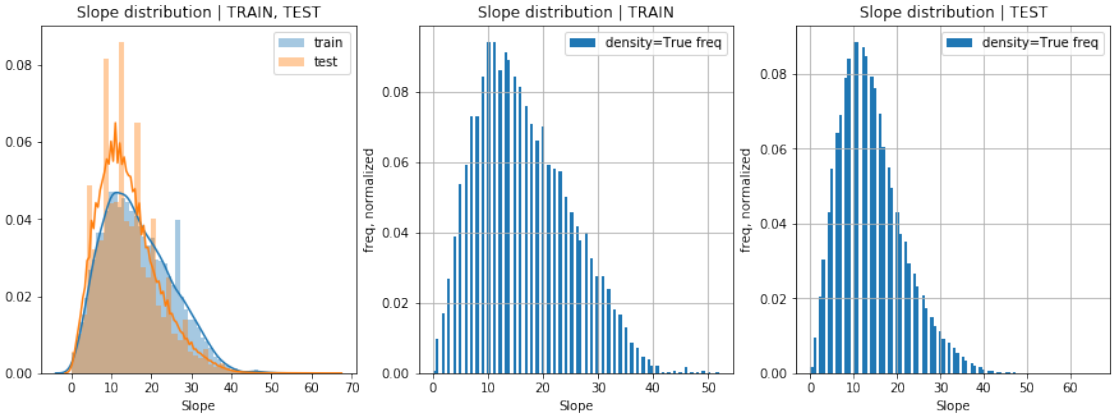
[9]:
```
# Remove redundant (for eda) features

train_target = train_df['Cover_Type']
train_df = train_df.drop( ['Id', 'Cover_Type'], axis=1 )

test_df = test_df.drop( ['Id'], axis=1 )
```

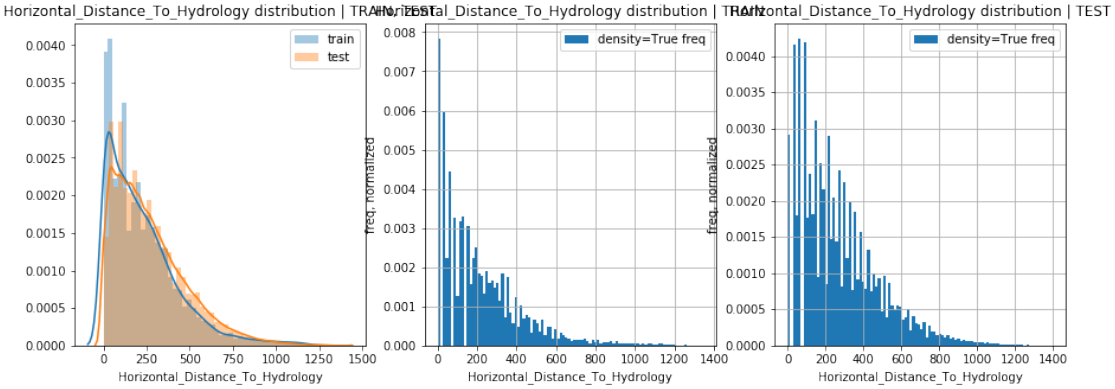
[74]:
```
def overview_distribution( df, col_name, ax, \
                          title_text='', n_bins=100, display_kde=False,
                           **hist_kwargs):
    df[col_name].hist( bins=n_bins, ax=ax, label='density=True freq',␣
 ↪density=True, **hist_kwargs )
    if display_kde:
        df[col_name].plot.kde( ax=ax, color='red', label='kde' )
    ax.set_xlabel(col_name)
    ax.set_ylabel('freq, normalized')
    ax.grid(True)
    ax.legend()
    ax.set_title('{0} distribution {1}'.format(col_name, title_text))


def overview_traintest_distributions( train_df, test_df, col_name, axes ):
    sns.distplot( train_df[col_name], ax=axes[0], label='train' )
    sns.distplot( test_df[col_name], ax=axes[0], label='test' )
    axes[0].set_title('{0} distribution | TRAIN, TEST'.format(col_name))
    axes[0].legend()
    overview_distribution( train_df, col_name, ax=axes[1], title_text='| TRAIN'␣
 ↪)
    overview_distribution( test_df, col_name, ax=axes[2], title_text='| TEST' )


def quick_traintest_distr_overview( train_df, test_df, col_name,␣
 ↪figsize_tuple=(15, 5) ):
    fig, axes = plt.subplots( 1, 3, figsize=figsize_tuple)
    overview_traintest_distributions( train_df, test_df, col_name, axes=axes )
```

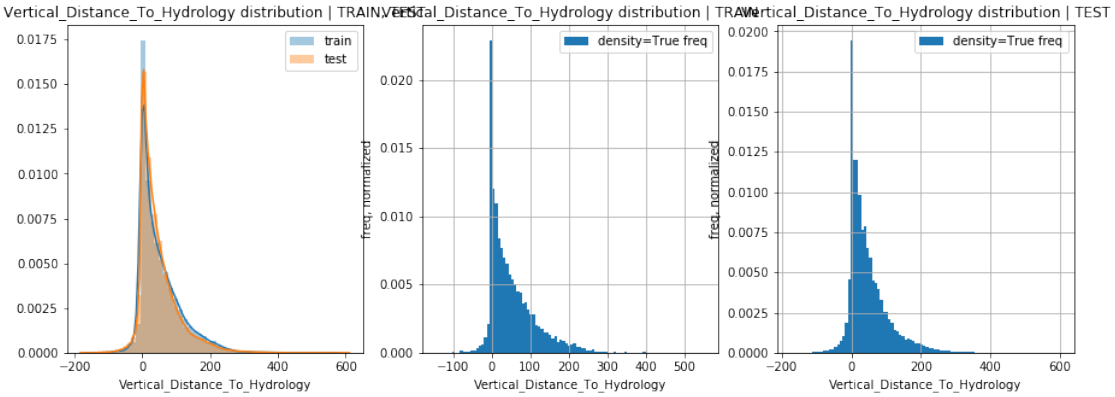
[84]:
```
# Overview distributions

for col_name in train_df.columns.values:
    quick_traintest_distr_overview( train_df, test_df, col_name )
```

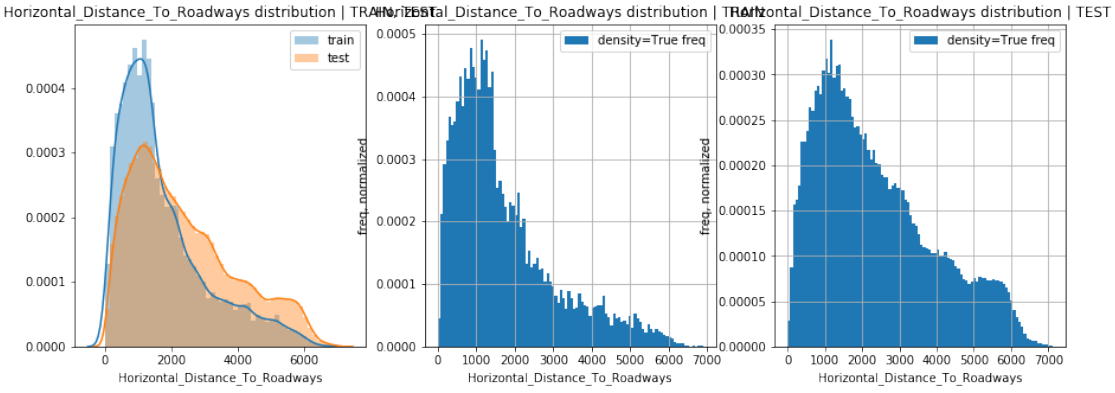

Elevation distribution | TRAIN, TEST
train
test
Elevation distribution | TRAIN
density=True freq
Elevation distribution | TEST
density=True freq
freq, normalized
Elevation

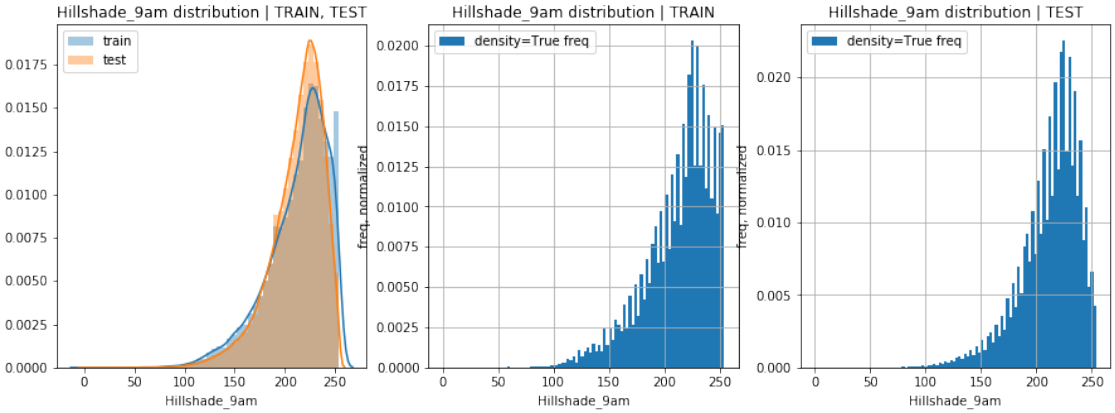

Aspect distribution | TRAIN, TEST
train
test
Aspect distribution | TRAIN
density=True freq
Aspect distribution | TEST
density=True freq
freq, normalized
Aspect

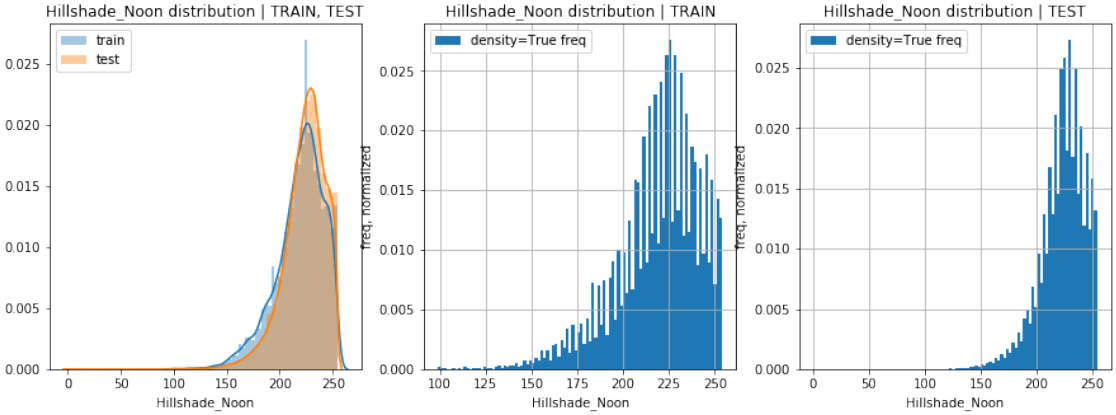

Slope distribution | TRAIN, TEST
train
test
Slope distribution | TRAIN
density=True freq
Slope distribution | TEST
density=True freq
freq, normalized
Slope

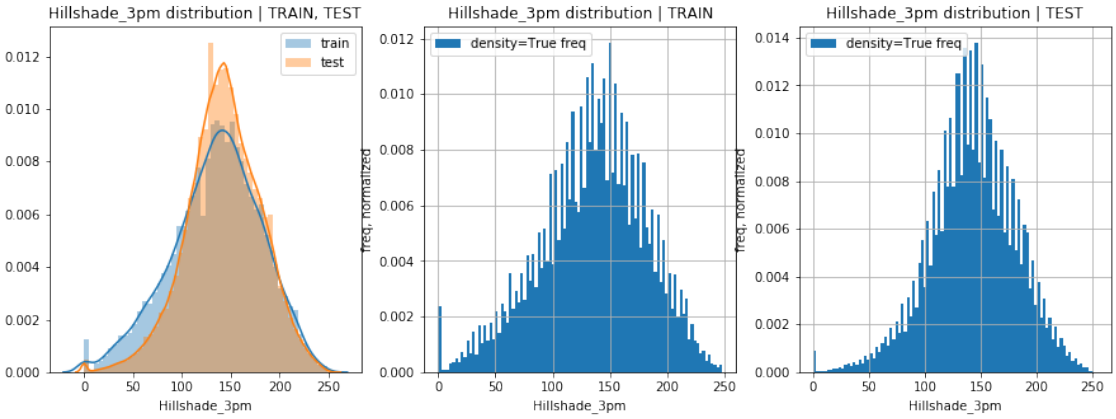

Horizontal_Distance_To_Hydrology distribution | TRAIN, TEST
train
test
Horizontal_Distance_To_Hydrology
Horizontal_Distance_To_Hydrology distribution | TRAIN
density=True freq
freq, normalized
Horizontal_Distance_To_Hydrology
Horizontal_Distance_To_Hydrology distribution | TEST
density=True freq
Horizontal_Distance_To_Hydrology

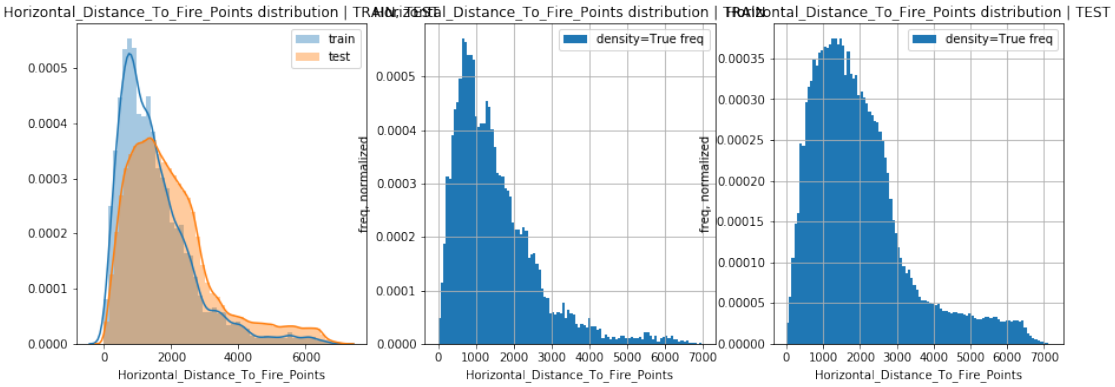

Vertical_Distance_To_Hydrology distribution | TRAIN, TEST
train
test
Vertical_Distance_To_Hydrology
Vertical_Distance_To_Hydrology distribution | TRAIN
density=True freq
freq, normalized
Vertical_Distance_To_Hydrology
Vertical_Distance_To_Hydrology distribution | TEST
density=True freq
Vertical_Distance_To_Hydrology

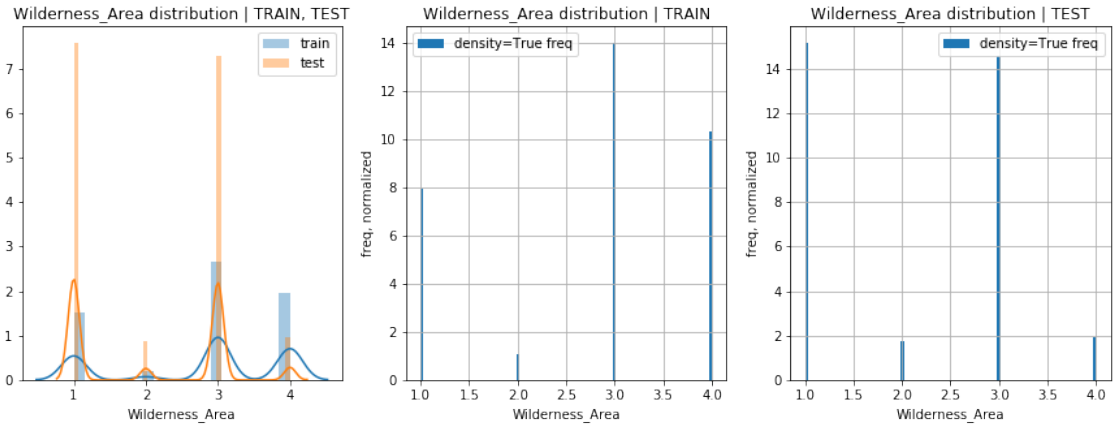

Horizontal_Distance_To_Roadways distribution | TRAIN, TEST
train
test
Horizontal_Distance_To_Roadways
Horizontal_Distance_To_Roadways distribution | TRAIN
density=True freq
freq, normalized
Horizontal_Distance_To_Roadways
Horizontal_Distance_To_Roadways distribution | TEST
density=True freq
Horizontal_Distance_To_Roadways

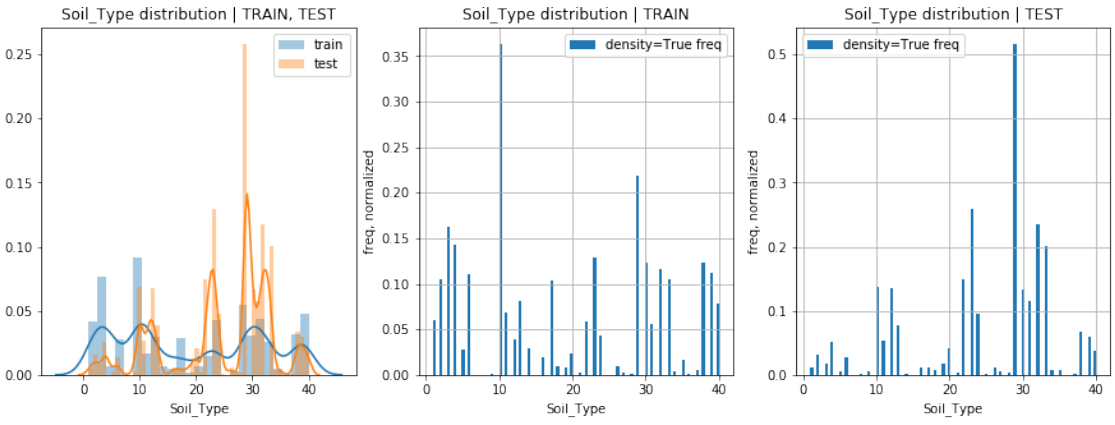

Hillshade_9am distribution | TRAIN, TEST
train
test
Hillshade_9am
Hillshade_9am distribution | TRAIN
density=True freq
freq, normalized
Hillshade_9am
Hillshade_9am distribution | TEST
density=True freq
Hillshade_9am


Hillshade_Noon distribution | TRAIN, TEST
train
test
Hillshade_Noon
Hillshade_Noon distribution | TRAIN
density=True freq
freq, normalized
Hillshade_Noon
Hillshade_Noon distribution | TEST
density=True freq
Hillshade_Noon


Hillshade_3pm distribution | TRAIN, TEST
train
test
Hillshade_3pm
Hillshade_3pm distribution | TRAIN
density=True freq
freq, normalized
Hillshade_3pm
Hillshade_3pm distribution | TEST
density=True freq
Hillshade_3pm

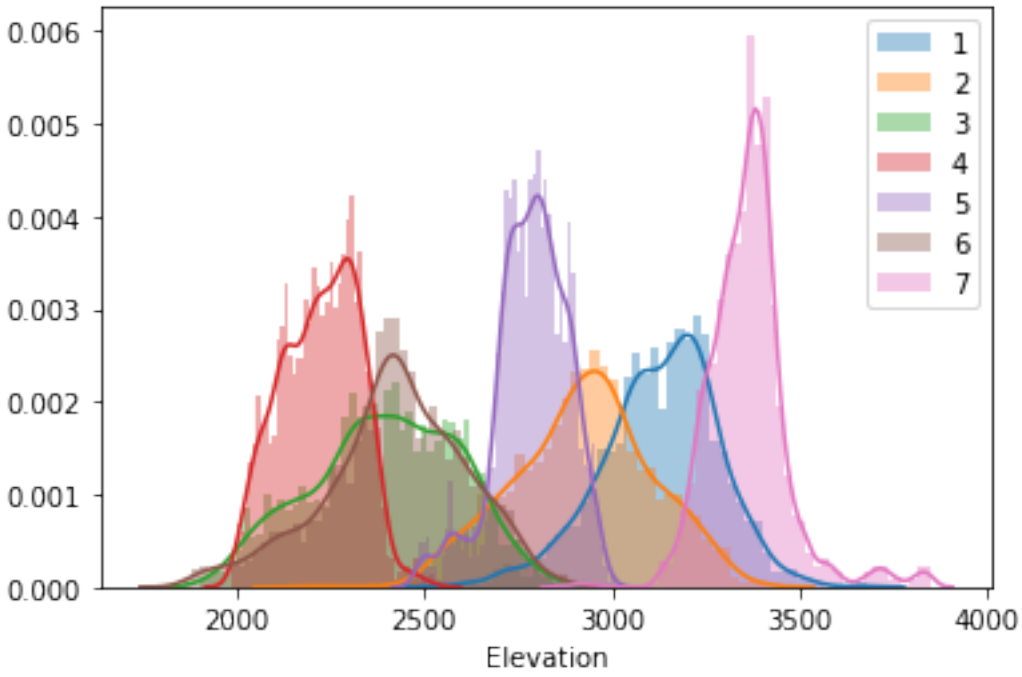


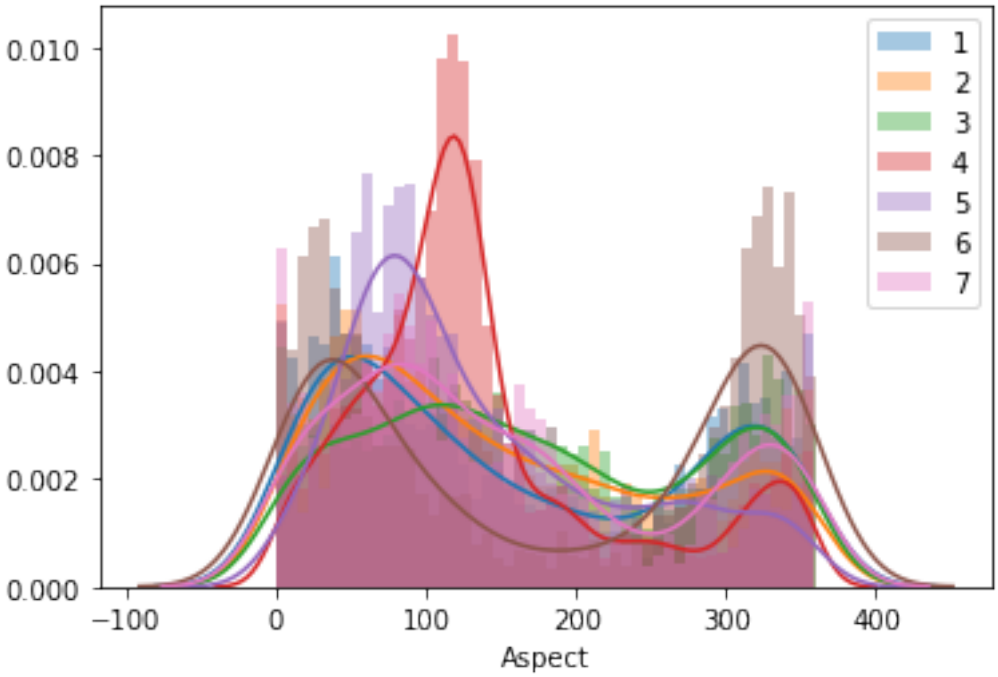



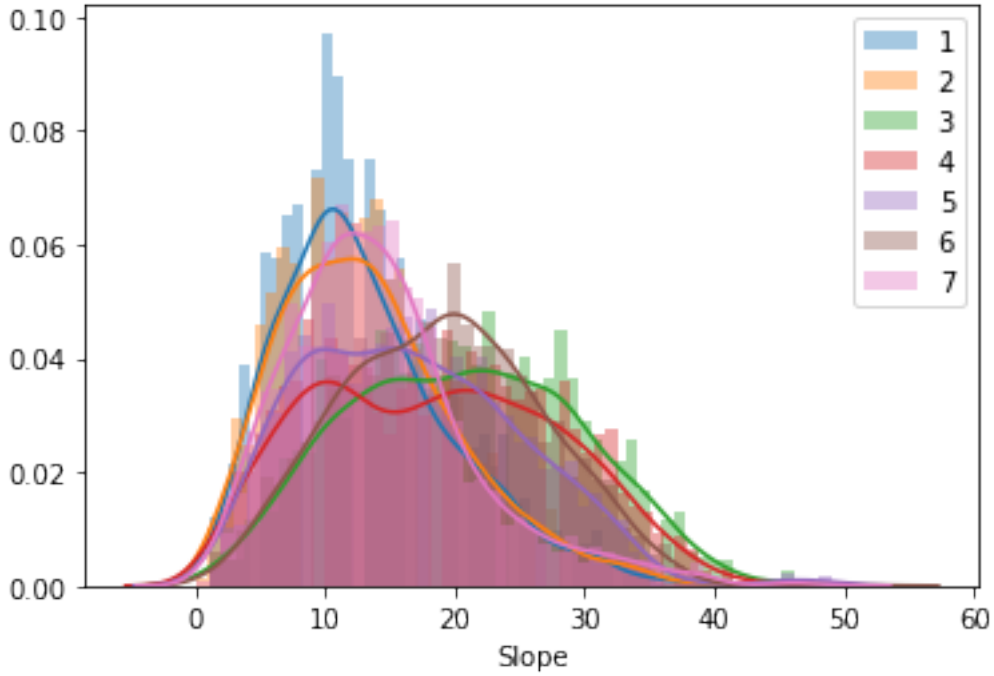



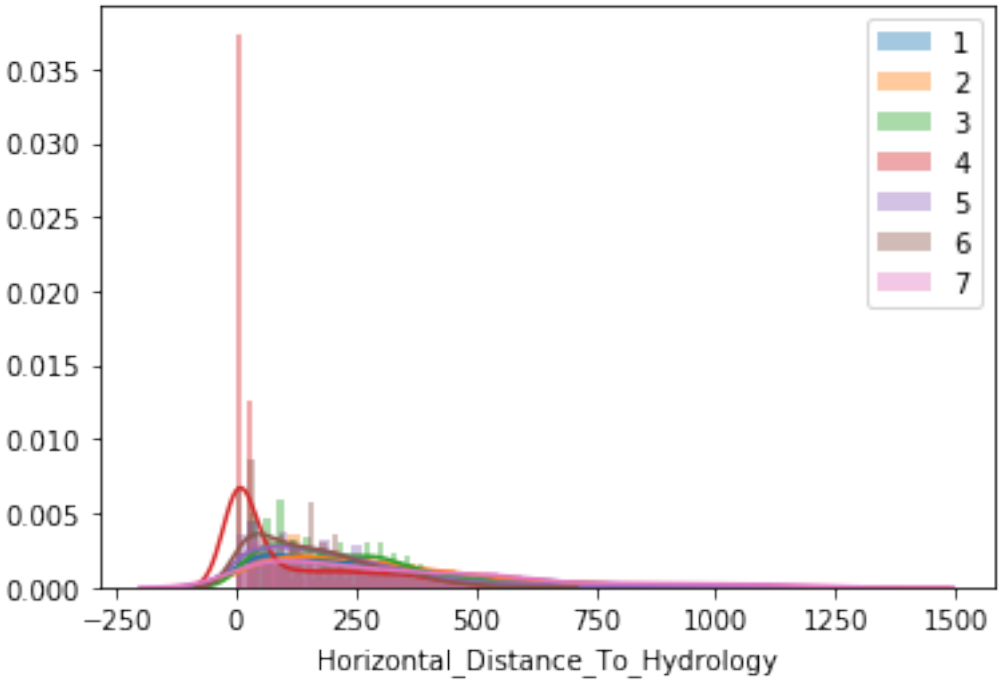

[109]:
```
# Overview distributions relative to target type

train_df['Cover_Type'] = train_target
```

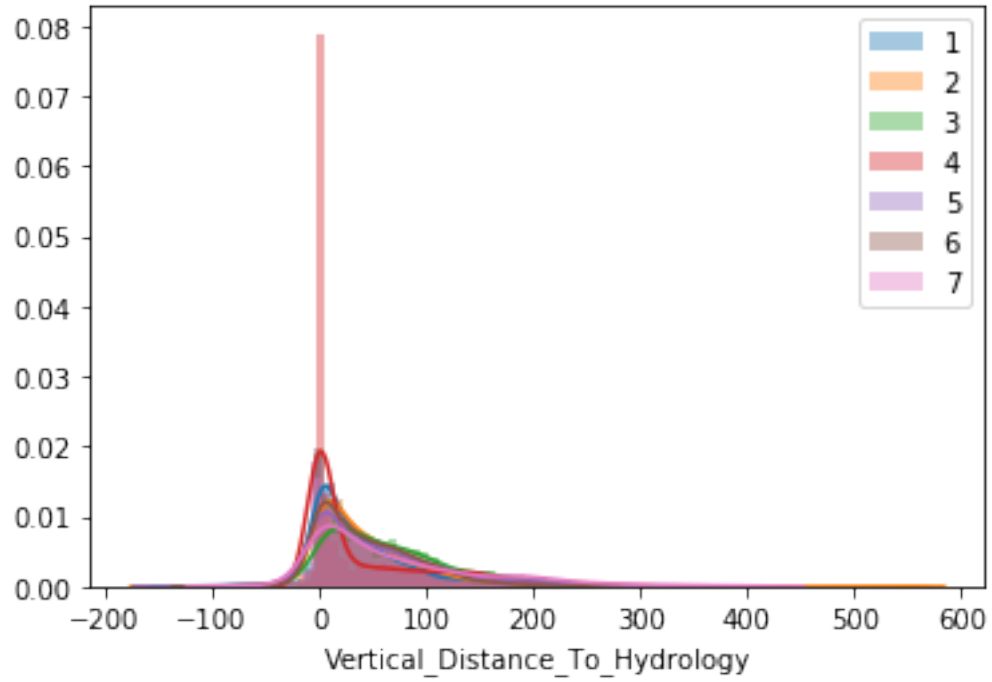
```
for col_name in train_df.columns.values[:-1]:
    for cover_type in sorted( train_df['Cover_Type'].unique() ):
        sns.distplot(
            train_df[ train_df['Cover_Type'] == cover_type ][col_name],
            label=cover_type,
            bins=50
        )
    plt.legend()
    plt.show()
```

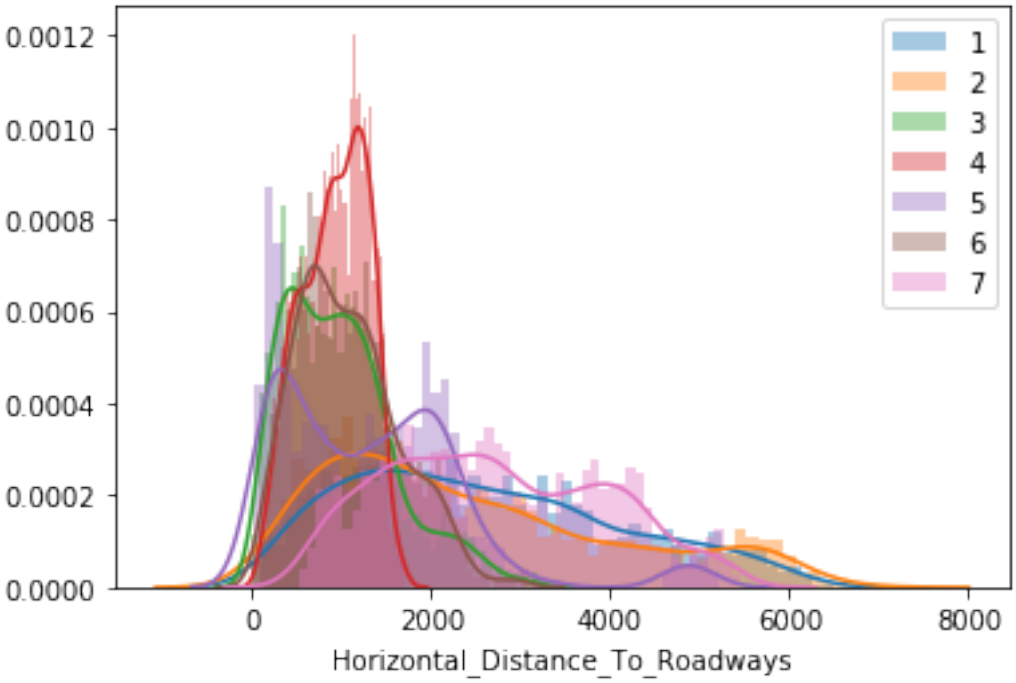


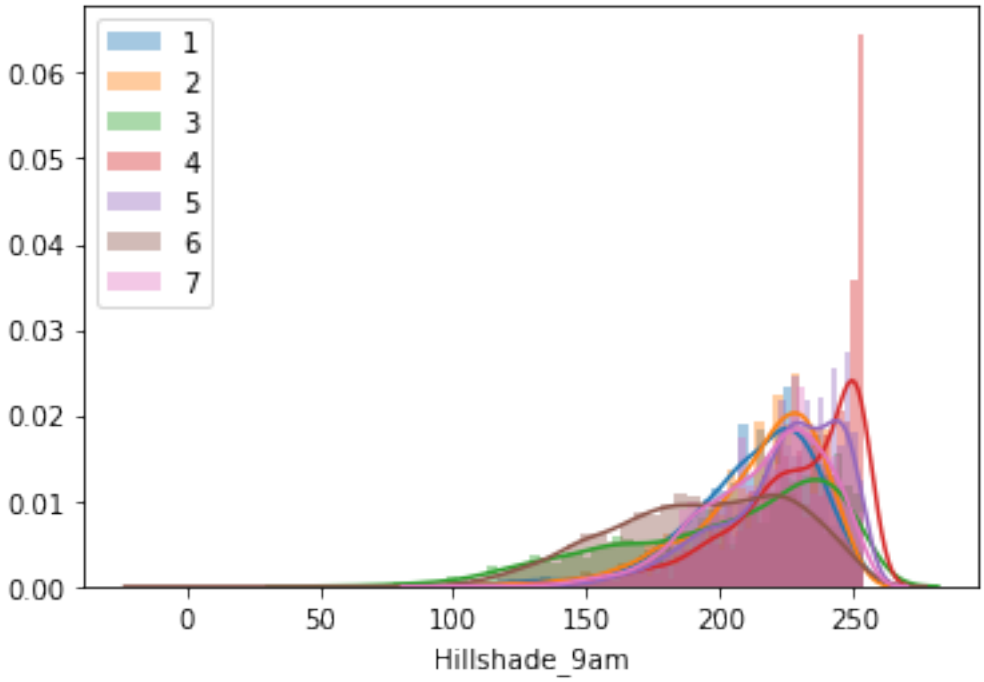

0.010
0.008
0.006
0.004
0.002
0.000
−100
0
100
200
300
400
Aspect
1
2
3
4
5
6
7

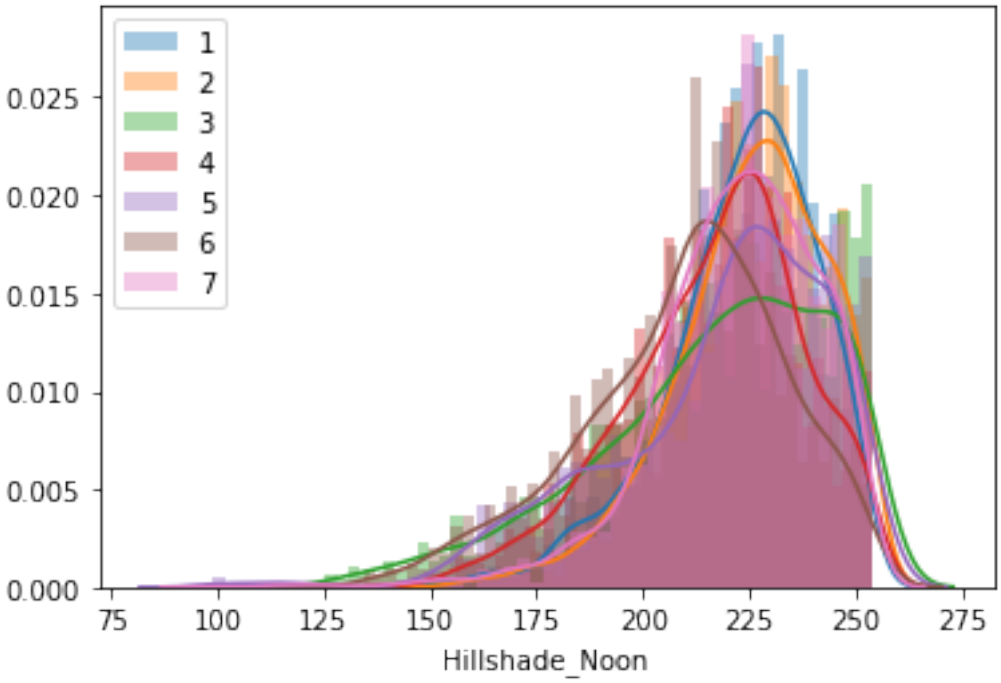

0.10
0.08
0.06
0.04
0.02
0.00
0
10
20
30
40
50
60
Slope
1
2
3
4
5
6
7

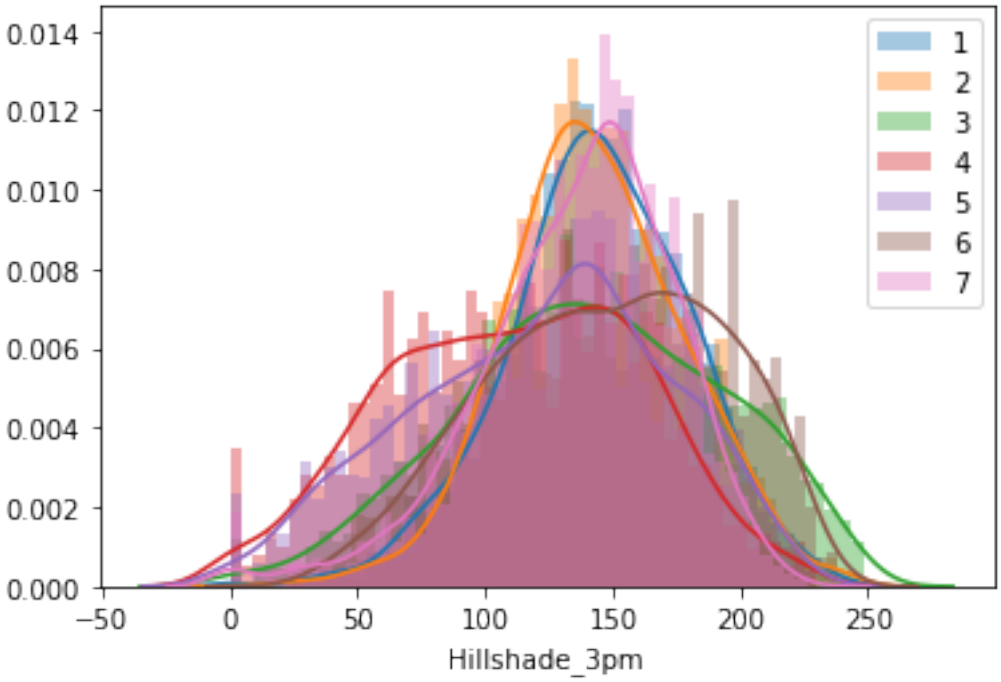

0.035
0.030
0.025
0.020
0.015
0.010
0.005
0.000
−250
0
250
500
750
1000
1250
1500
Horizontal_Distance_To_Hydrology
1
2
3
4
5
6
7

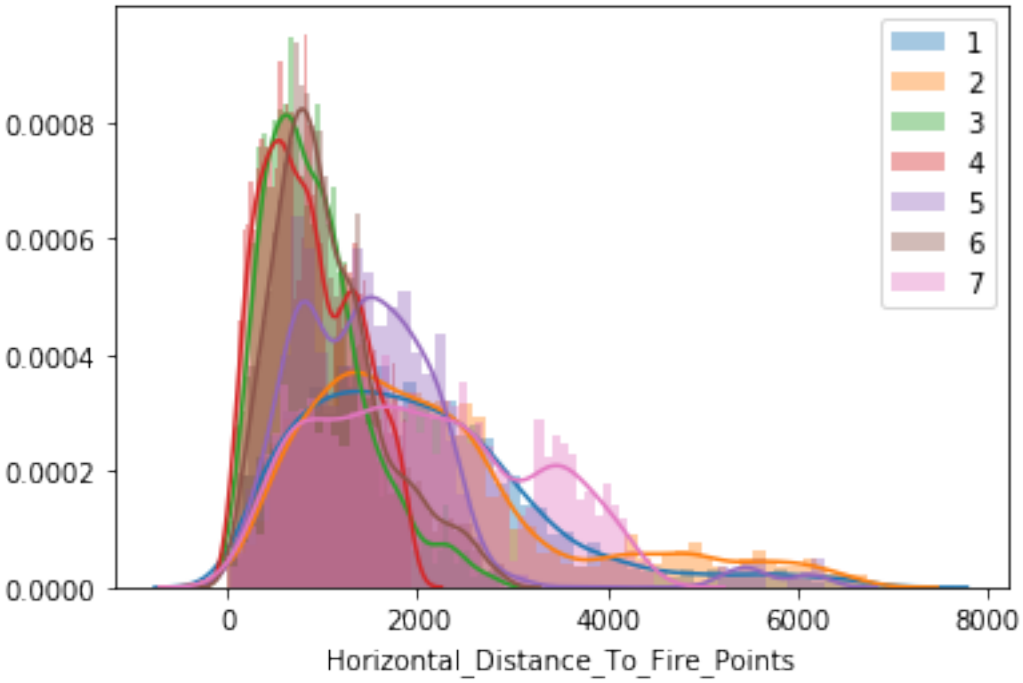

0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
−200
−100
0
100
200
300
400
500
600
Vertical_Distance_To_Hydrology
1
2
3
4
5
6
7

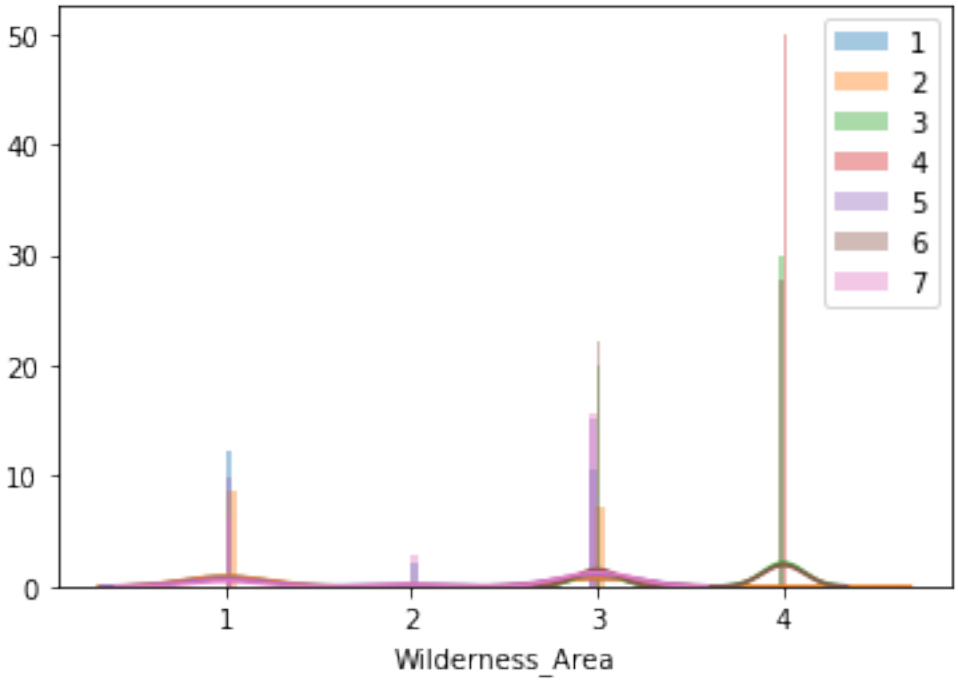

0.0012
0.0010
0.0008
0.0006
0.0004
0.0002
0.0000
0
2000
4000
6000
8000
Horizontal_Distance_To_Roadways
1
2
3
4
5
6
7

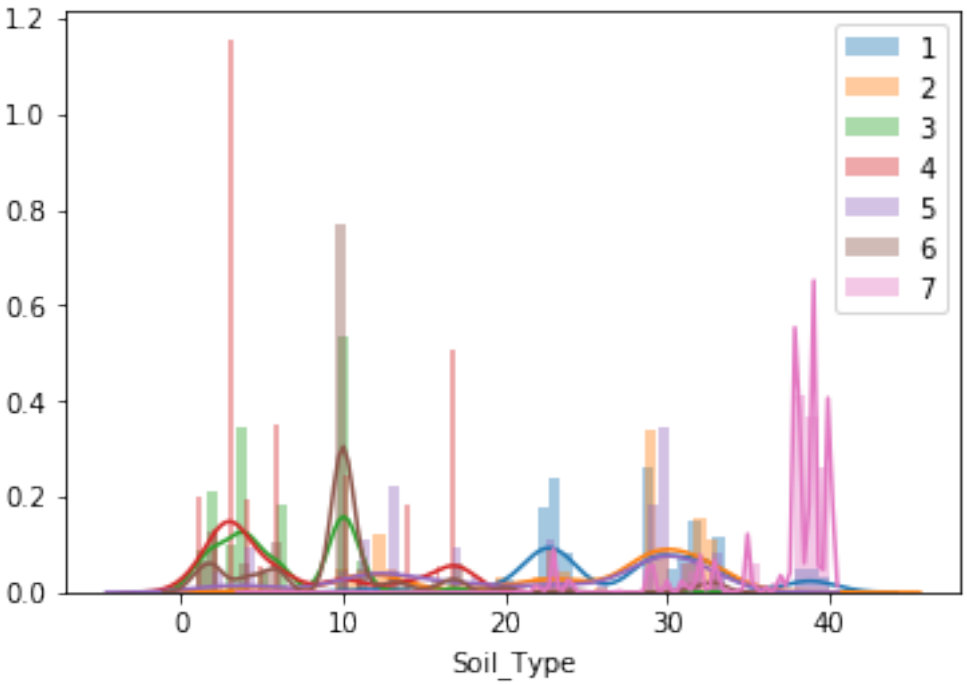

0.06
0.05
0.04
0.03
0.02
0.01
0.00
0
50
100
150
200
250
Hillshade_9am
1
2
3
4
5
6
7


1
2
3
4
5
6
7
0.025
0.020
0.015
0.010
0.005
0.000
75
100
125
150
175
200
225
250
275
Hillshade_Noon

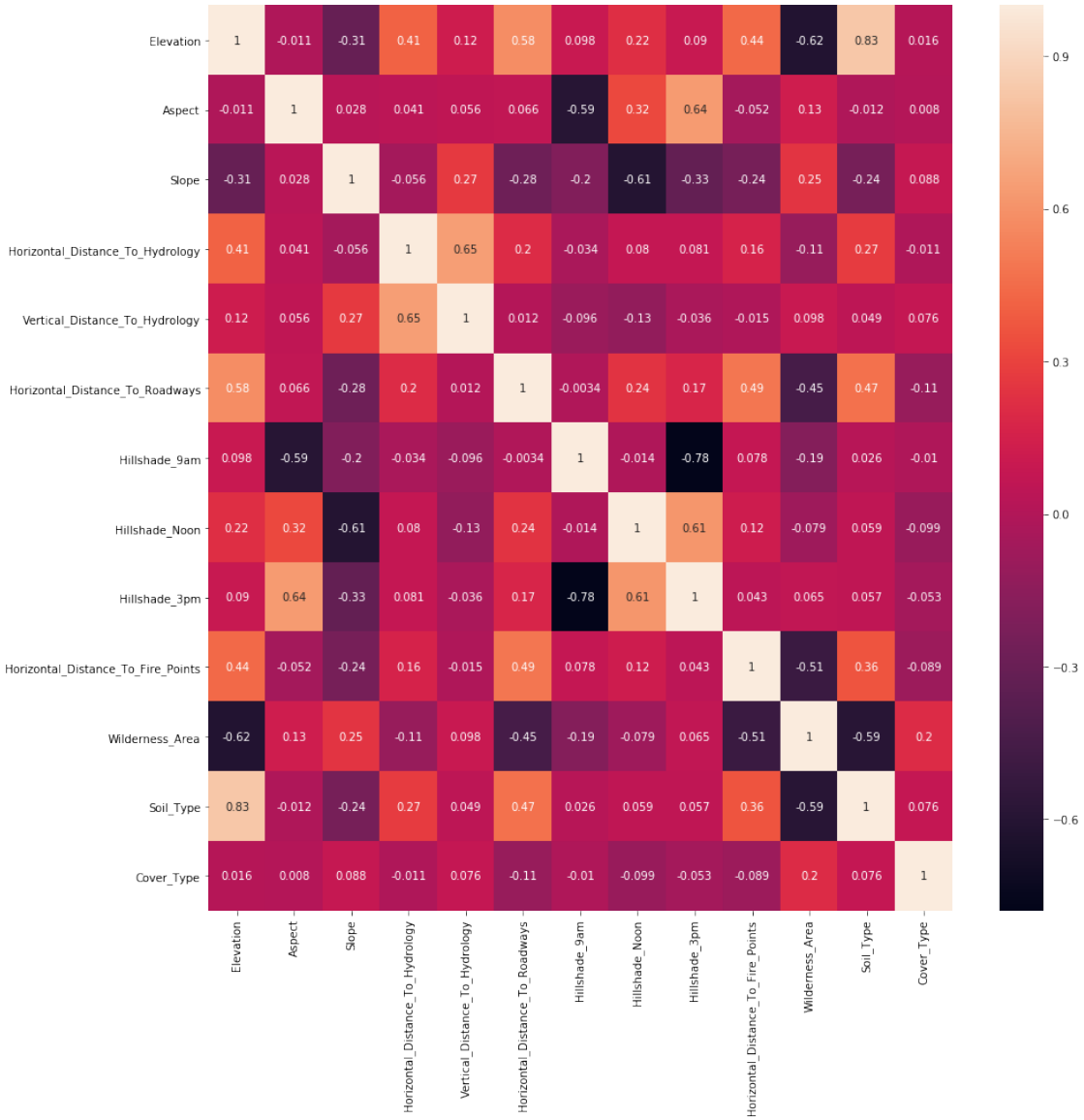

1
2
3
4
5
6
7
0.014
0.012
0.010
0.008
0.006
0.004
0.002
0.000
−50
0
50
100
150
200
250
Hillshade_3pm

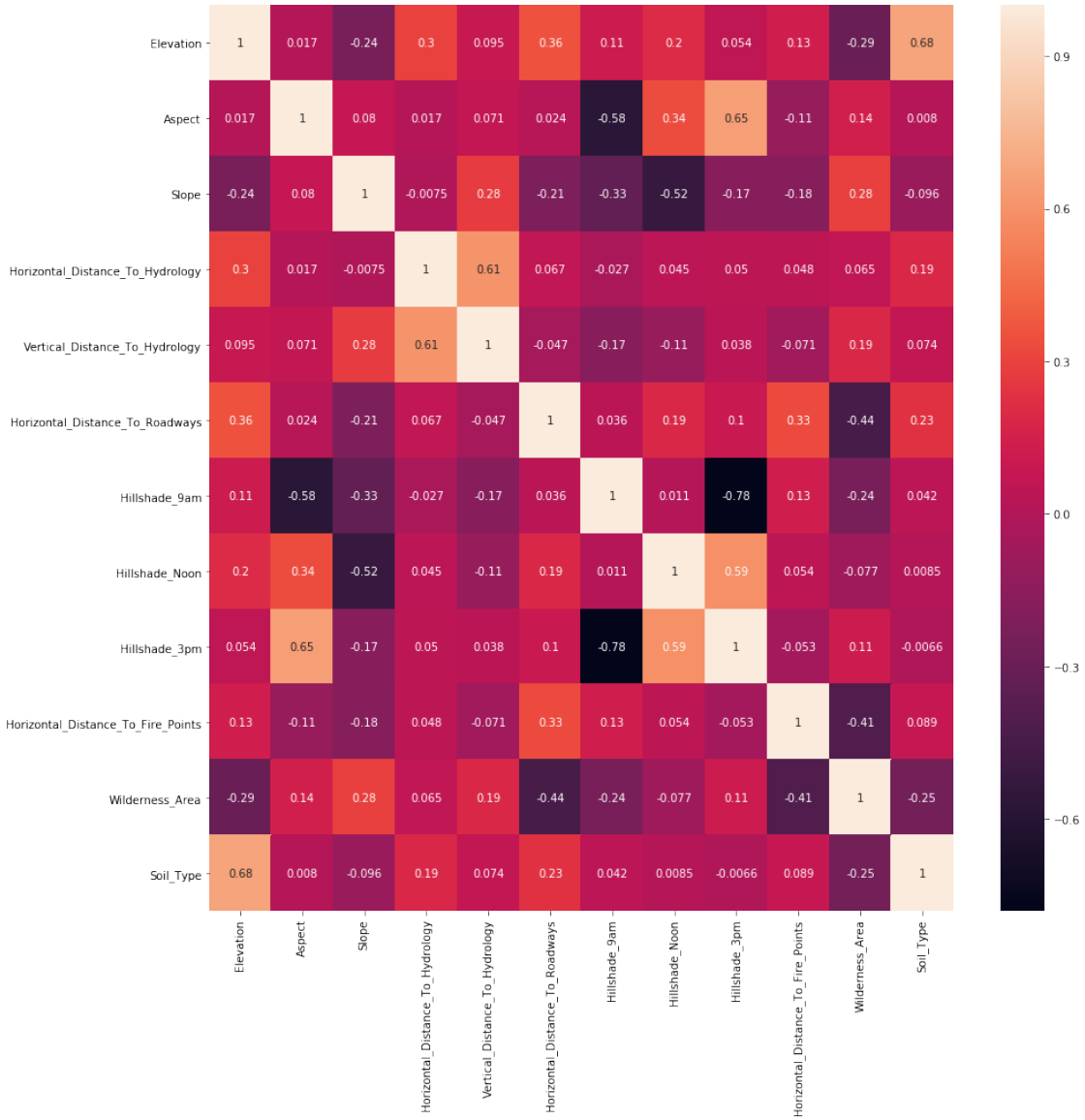

1
2
3
4
5
6
7
0.0008
0.0006
0.0004
0.0002
0.0000
0
2000
4000
6000
8000
Horizontal_Distance_To_Fire_Points


1
2
3
4
5
6
7
50
40
30
20
10
0
1
2
3
4
Wilderness_Area




[95]:
```
# Overview correlation in train / test sets

def overview_pearson_corr( df ):
    fig = plt.figure( figsize=(15, 15) )
    sns.heatmap(
        df.corr(),
        annot=True
    )
    plt.autoscale()
    plt.show()

train_df['Cover_Type'] = train_target
overview_pearson_corr( train_df )

overview_pearson_corr( test_df )
```


Elevation 1 -0.011 -0.31 0.41 0.12 0.58 0.098 0.22 0.09 0.44 -0.62 0.83 0.016
Aspect -0.011 1 0.028 0.041 0.056 0.066 -0.59 0.32 0.64 -0.052 0.13 -0.012 0.008
Slope -0.31 0.028 1 -0.056 0.27 -0.28 -0.2 -0.61 -0.33 -0.24 0.25 -0.24 0.088
Horizontal_Distance_To_Hydrology 0.41 0.041 -0.056 1 0.65 0.2 -0.034 0.08 0.081 0.16 -0.11 0.27 -0.011
Vertical_Distance_To_Hydrology 0.12 0.056 0.27 0.65 1 0.012 -0.096 -0.13 -0.036 -0.015 0.098 0.049 0.076
Horizontal_Distance_To_Roadways 0.58 0.066 -0.28 0.2 0.012 1 -0.0034 0.24 0.17 0.49 -0.45 0.47 -0.11
Hillshade_9am 0.098 -0.59 -0.2 -0.034 -0.096 -0.0034 1 -0.014 -0.78 0.078 -0.19 0.026 -0.01
Hillshade_Noon 0.22 0.32 -0.61 0.08 -0.13 0.24 -0.014 1 0.61 0.12 -0.079 0.059 -0.099
Hillshade_3pm 0.09 0.64 -0.33 0.081 -0.036 0.17 -0.78 0.61 1 0.043 0.065 0.057 -0.053
Horizontal_Distance_To_Fire_Points 0.44 -0.052 -0.24 0.16 -0.015 0.49 0.078 0.12 0.043 1 -0.51 0.36 -0.089
Wilderness_Area -0.62 0.13 0.25 -0.11 0.098 -0.45 -0.19 -0.079 0.065 -0.51 1 -0.59 0.2
Soil_Type 0.83 -0.012 -0.24 0.27 0.049 0.47 0.026 0.059 0.057 0.36 -0.59 1 0.076
Cover_Type 0.016 0.008 0.088 -0.011 0.076 -0.11 -0.01 -0.099 -0.053 -0.089 0.2 0.076 1
Elevation
Aspect
Slope
Horizontal_Distance_To_Hydrology
Vertical_Distance_To_Hydrology
Horizontal_Distance_To_Roadways
Hillshade_9am
Hillshade_Noon
Hillshade_3pm
Horizontal_Distance_To_Fire_Points
Wilderness_Area
Soil_Type
Cover_Type
0.9
0.6
0.3
0.0
−0.3
−0.6



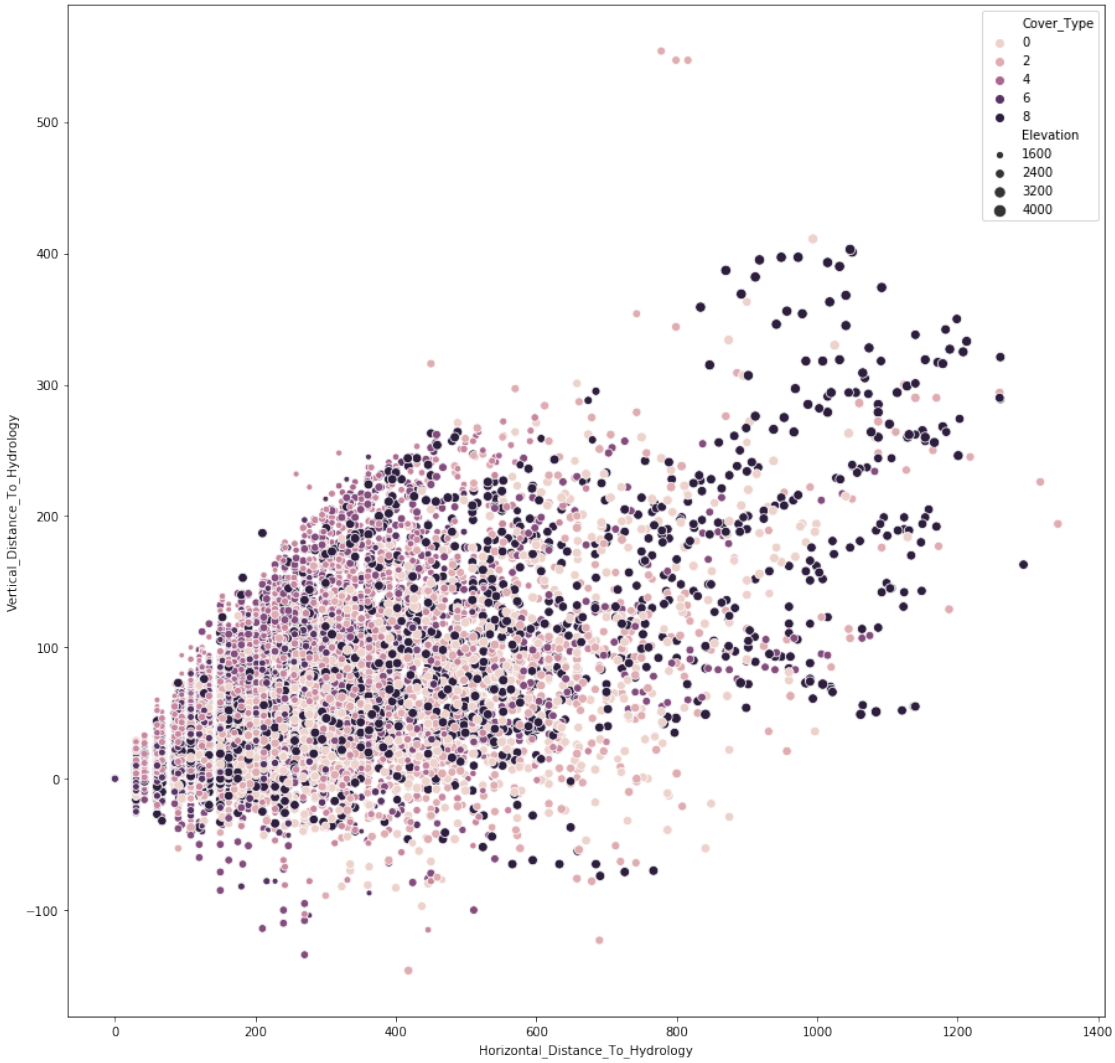

[126]:
```
# Skewnewss

def overview_bad_kurtosis_skewness( df, df_str_descr ):
    # kurtosis
    print('{0}: Features, where kurtosis not as in normal univariate␣
 ↪distribution:'.format(df_str_descr))
    df_kurt = df.kurtosis()
    display(
        df_kurt[ (df_kurt < -2) | (df_kurt > 2) ]
    )
    # skew
```

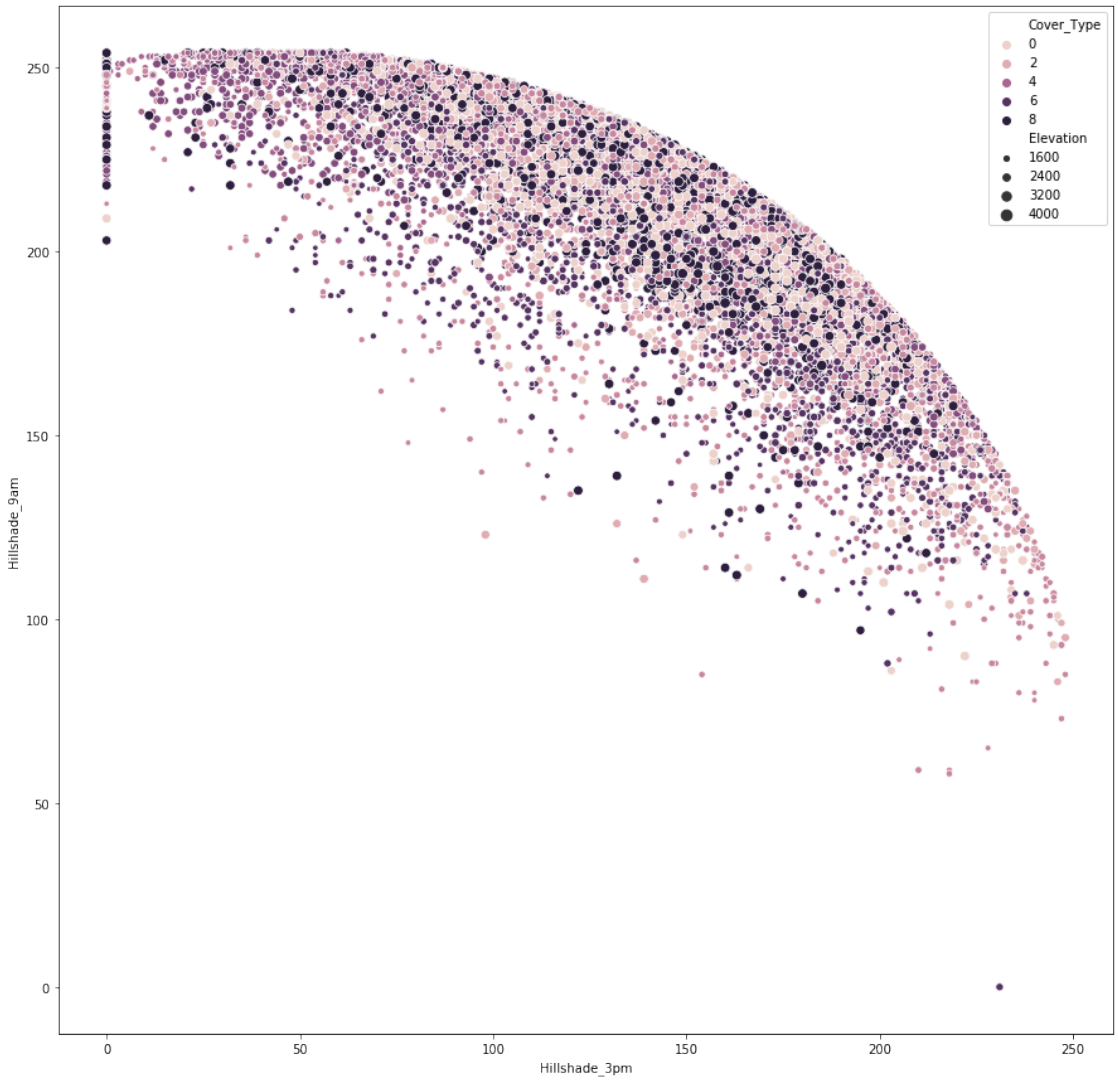
```
    print('{0}: Features, where skew not as in normal univariate distribution:'.
↪format(df_str_descr))
    df_skewness = df.skew()
    display(
        df_skewness[ (df_skewness < -1) | (df_skewness > 1) ]
    )


overview_bad_kurtosis_skewness( train_df, 'TRAIN' )
overview_bad_kurtosis_skewness( test_df, 'TEST' )
```

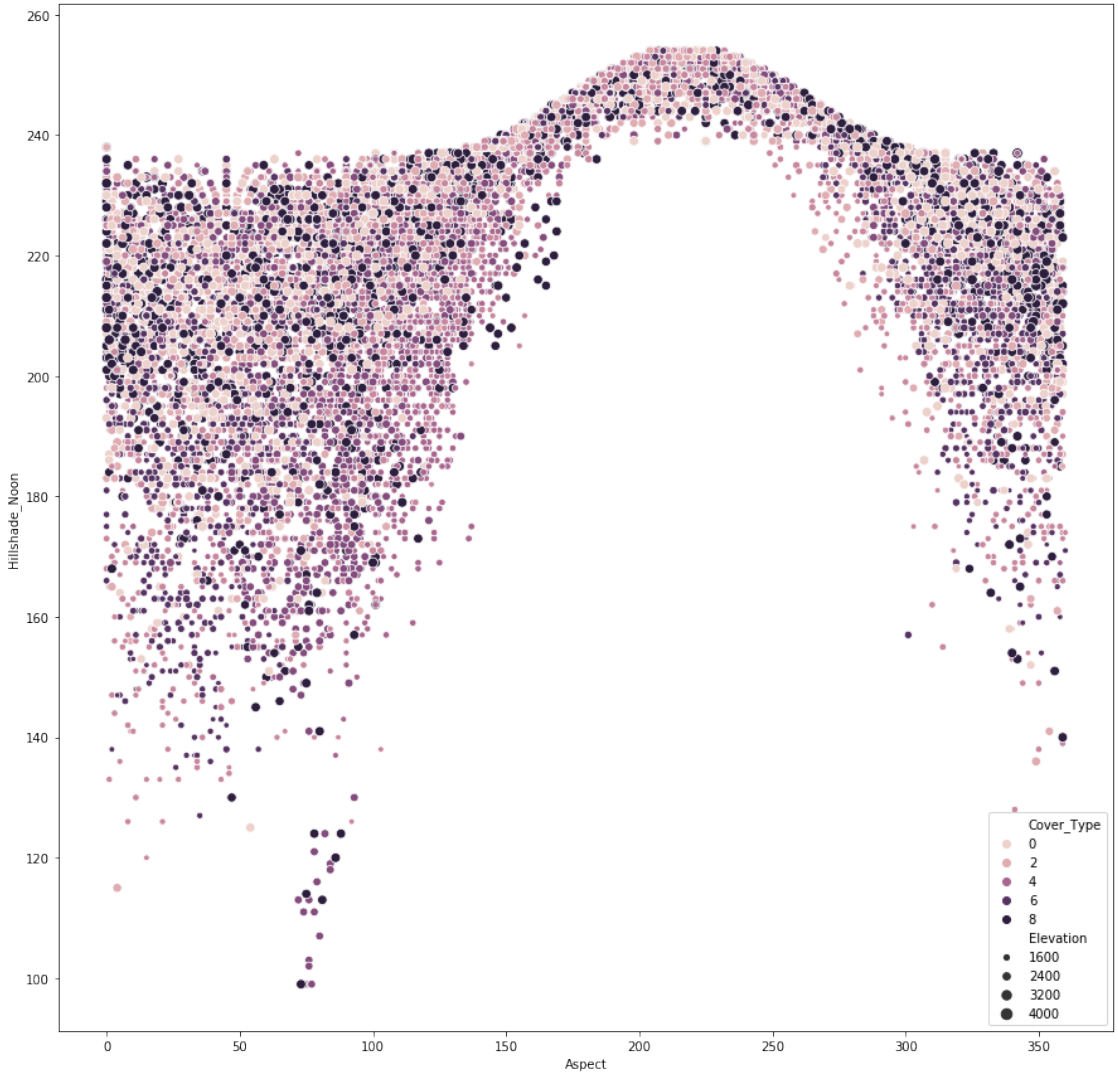
```
TRAIN: Features, where kurtosis not as in normal univariate distribution:

Horizontal_Distance_To_Hydrology      2.803984
Vertical_Distance_To_Hydrology        3.403499
Horizontal_Distance_To_Fire_Points    3.385416
dtype: float64


TRAIN: Features, where skew not as in normal univariate distribution:

Horizontal_Distance_To_Hydrology      1.488052
Vertical_Distance_To_Hydrology        1.537776
Horizontal_Distance_To_Roadways       1.247811
Hillshade_9am                        -1.093681
Horizontal_Distance_To_Fire_Points    1.617099
dtype: float64


TEST: Features, where kurtosis not as in normal univariate distribution:

Vertical_Distance_To_Hydrology    5.310146
Hillshade_Noon                    2.087615
dtype: float64


TEST: Features, where skew not as in normal univariate distribution:

Horizontal_Distance_To_Hydrology      1.133163
Vertical_Distance_To_Hydrology        1.797687
Hillshade_9am                        -1.184138
Hillshade_Noon                       -1.062230
Horizontal_Distance_To_Fire_Points    1.281245
dtype: float64
```

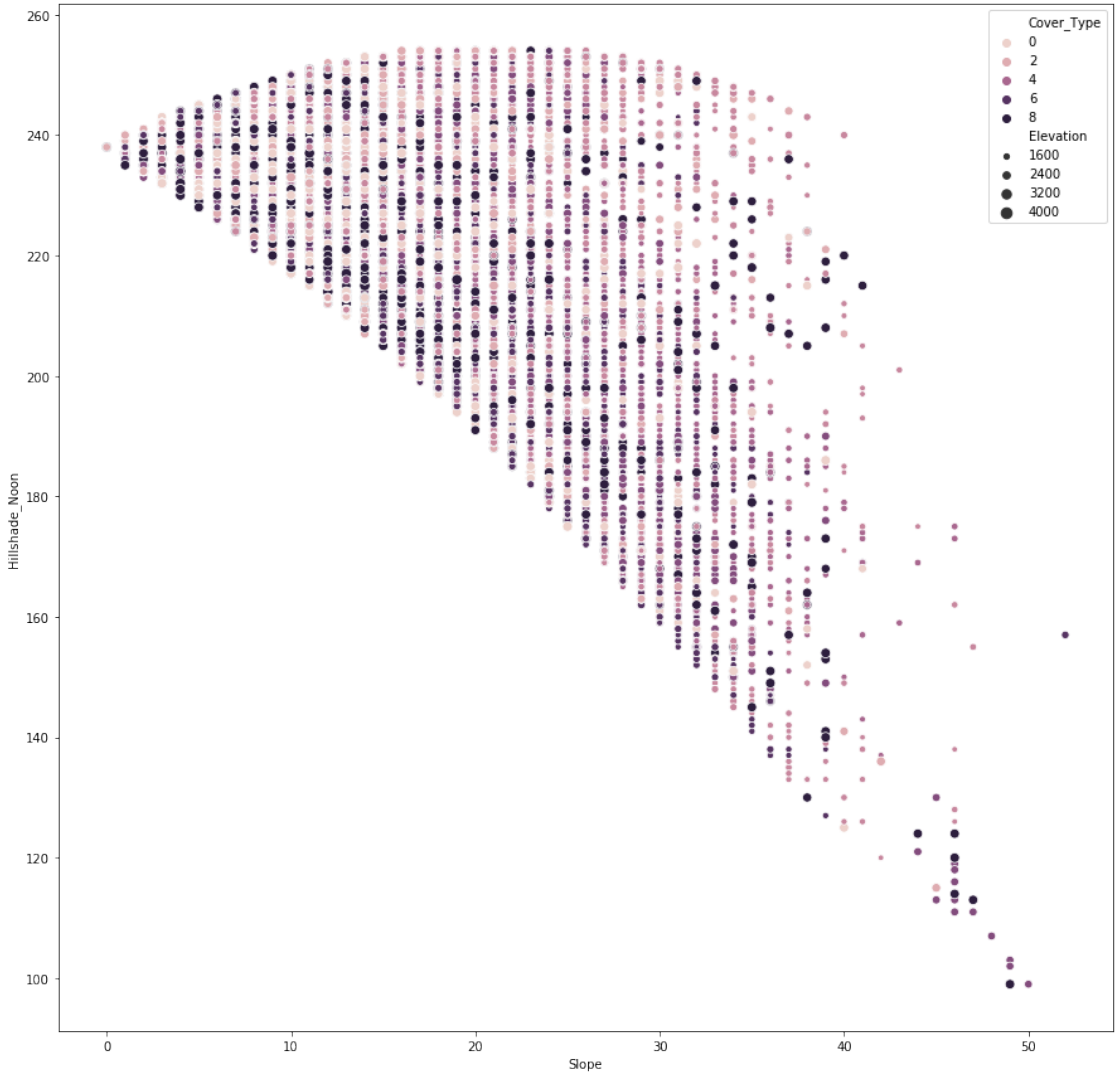
[162]:
```
# Overview scatter plots relative to target type
```

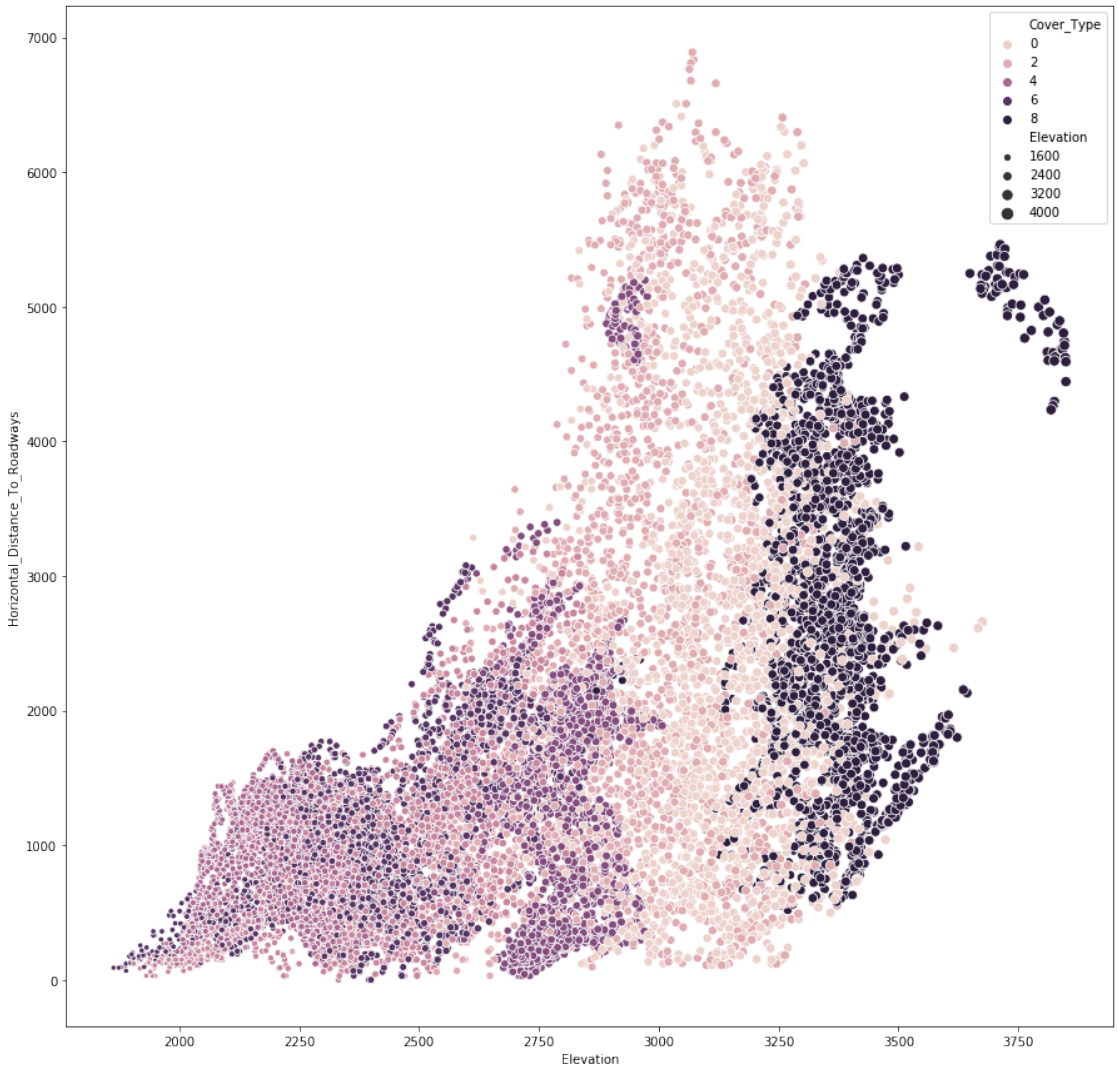
```
def overview_scatter_x_y_color_covertype( df, x, y, hue='Cover_Type',␣
 ↪size='Elevation' ):
    fig = plt.figure(figsize=(15, 15))
    sns.scatterplot(
        x=x, y=y,
        hue=hue,
        size=size,
        data=df,
    )
    plt.show()

overview_scatter_x_y_color_covertype(
    train_df, 'Horizontal_Distance_To_Hydrology',␣
 ↪'Vertical_Distance_To_Hydrology'
)

overview_scatter_x_y_color_covertype(
    train_df, 'Hillshade_3pm', 'Hillshade_9am'
)

overview_scatter_x_y_color_covertype(
    train_df, 'Aspect', 'Hillshade_Noon'
)

overview_scatter_x_y_color_covertype(
    train_df, 'Slope', 'Hillshade_Noon'
)

overview_scatter_x_y_color_covertype(
    train_df, 'Elevation', 'Horizontal_Distance_To_Roadways'
)
```


Cover_Type
0
2
4
6
8
Elevation
1600
2400
3200
4000
Vertical_Distance_To_Hydrology
500
400
300
200
100
0
−100
0
200
400
600
800
1000
1200
1400
Horizontal_Distance_To_Hydrology


Cover_Type
0
2
4
6
8
Elevation
1600
2400
3200
4000
Hillshade_9am
Hillshade_3pm
0
50
100
150
200
250


Hillshade_Noon
Aspect
260
240
220
200
180
160
140
120
100
0
50
100
150
200
250
300
350
Cover_Type
0
2
4
6
8
Elevation
1600
2400
3200
4000


Cover_Type
0
2
4
6
8
Elevation
1600
2400
3200
4000
Hillshade_Noon
Slope
260
240
220
200
180
160
140
120
100
0
10
20
30
40
50


Cover_Type
0
2
4
6
8
Elevation
1600
2400
3200
4000
Horizontal_Distance_To_Roadways
7000
6000
5000
4000
3000
2000
1000
0
2000
2250
2500
2750
3000
3250
3500
3750
Elevation