A2 Car Prices Assingment

1. Import Library

```
In [1]:   import numpy as np
          import pandas as pd
          import seaborn as sns
          import matplotlib.pyplot as plt
```

```
In [2]:   import matplotlib
          np.__version__, pd.__version__, sns.__version__, matplotlib.__version__
```

Out[2]:   ('2.0.2', '2.2.3', '0.13.2', '3.9.4')

2. Load Data

```
In [3]:   #Load Data
          df = pd.read_csv('Cars.csv')
```

```
In [4]:   #print the first rows of the data
          df.head()
```

Out[4]:

	name	year	selling_price	km_driven	fuel	seller_type	transmission	ow
0	Maruti Swift Dzire VDI	2014	450000	145500	Diesel	Individual	Manual	F Ow
1	Skoda Rapid 1.5 TDI Ambition	2014	370000	120000	Diesel	Individual	Manual	Seco Ow
2	Honda City 2017-2020 EXi	2006	158000	140000	Petrol	Individual	Manual	Th Ow
3	Hyundai i20 Sportz Diesel	2010	225000	127000	Diesel	Individual	Manual	F Ow
4	Maruti Swift VXI BSIII	2007	130000	120000	Petrol	Individual	Manual	F Ow

```
In [5]:   #Checking shape of the data
          df.shape
```

Out[5]:   (8128, 13)

In [6]: `#Statiscal Info of the data`
`df.describe()`

Out[6]:

	year	selling_price	km_driven	seats
count	8128.000000	8.128000e+03	8.128000e+03	7907.000000
mean	2013.804011	6.382718e+05	6.981951e+04	5.416719
std	4.044249	8.062534e+05	5.655055e+04	0.959588
min	1983.000000	2.999900e+04	1.000000e+00	2.000000
25%	2011.000000	2.549990e+05	3.500000e+04	5.000000
50%	2015.000000	4.500000e+05	6.000000e+04	5.000000
75%	2017.000000	6.750000e+05	9.800000e+04	5.000000
max	2020.000000	1.000000e+07	2.360457e+06	14.000000

In [7]: `#Check data type`
`df.info()`

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8128 entries, 0 to 8127
Data columns (total 13 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   name           8128 non-null   object
 1   year           8128 non-null   int64
 2   selling_price  8128 non-null   int64
 3   km_driven      8128 non-null   int64
 4   fuel           8128 non-null   object
 5   seller_type    8128 non-null   object
 6   transmission   8128 non-null   object
 7   owner          8128 non-null   object
 8   mileage        7907 non-null   object
 9   engine         7907 non-null   object
 10  max_power      7913 non-null   object
 11  torque         7906 non-null   object
 12  seats          7907 non-null   float64
dtypes: float64(1), int64(3), object(9)
memory usage: 825.6+ KB
```

In [8]: `#Check column names`
`df.columns`

Out[8]: `Index(['name', 'year', 'selling_price', 'km_driven', 'fuel', 'seller_typ`
`e',`
`       'transmission', 'owner', 'mileage', 'engine', 'max_power', 'torqu`
`e',`
`       'seats'],`
`      dtype='object')`

3. Exploratory Data Analysis (EDA)

In [9]: `#Check column names`
`df.columns`

Out[9]: Index(['name', 'year', 'selling_price', 'km_driven', 'fuel', 'seller_typ
         e',
                'transmission', 'owner', 'mileage', 'engine', 'max_power', 'torqu
         e',
                'seats'],
              dtype='object')

In [10]:
```python
#renaming column 'name' to 'brand'
df.rename(columns= {
    'name': 'brand'}, inplace= True)

df.columns
```

Out[10]: Index(['brand', 'year', 'selling_price', 'km_driven', 'fuel', 'seller_ty
         pe',
                'transmission', 'owner', 'mileage', 'engine', 'max_power', 'torqu
         e',
                'seats'],
              dtype='object')
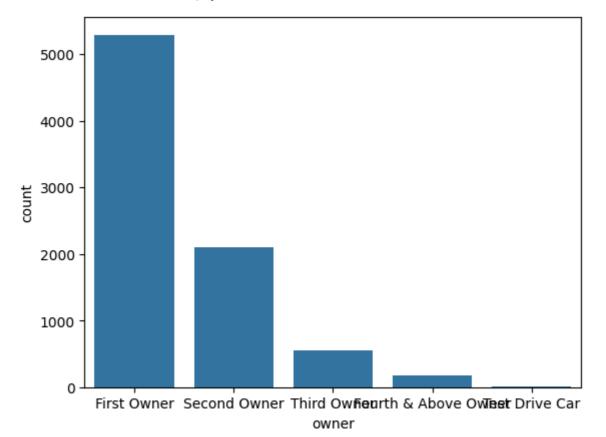
Univariate analyis

Count Plot

In [11]:
```python
sns.countplot(data = df, x = 'owner')
```

Out[11]: <Axes: xlabel='owner', ylabel='count'>
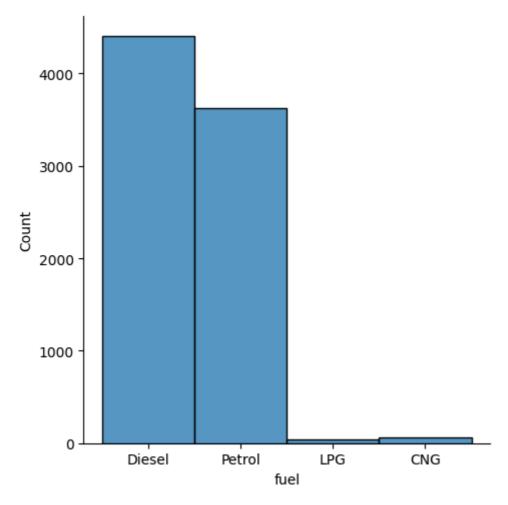

Dis Plot

In [12]:
```python
sns.displot(data = df, x = 'fuel')
```

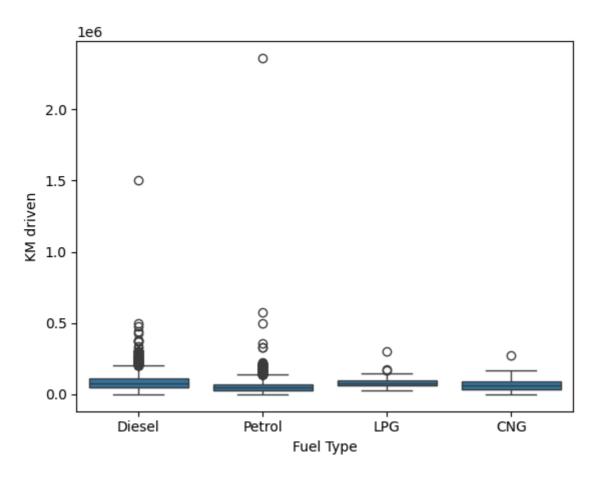Out[12]: <seaborn.axisgrid.FacetGrid at 0x13db65e80>

Multivariate analysis

Box Plot

```
In [13]: sns.boxplot(x = df["fuel"], y = df["km_driven"]);
         plt.ylabel("KM driven")
         plt.xlabel("Fuel Type")
```
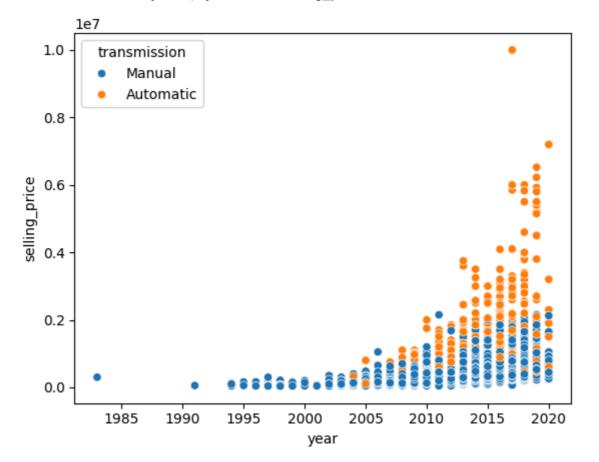
Out[13]: Text(0.5, 0, 'Fuel Type')

Scatter Plot

```
In [14]: sns.scatterplot(x = df['year'], y = df['selling_price'], hue=df['transmis
```

```
Out[14]: <Axes: xlabel='year', ylabel='selling_price'>
```

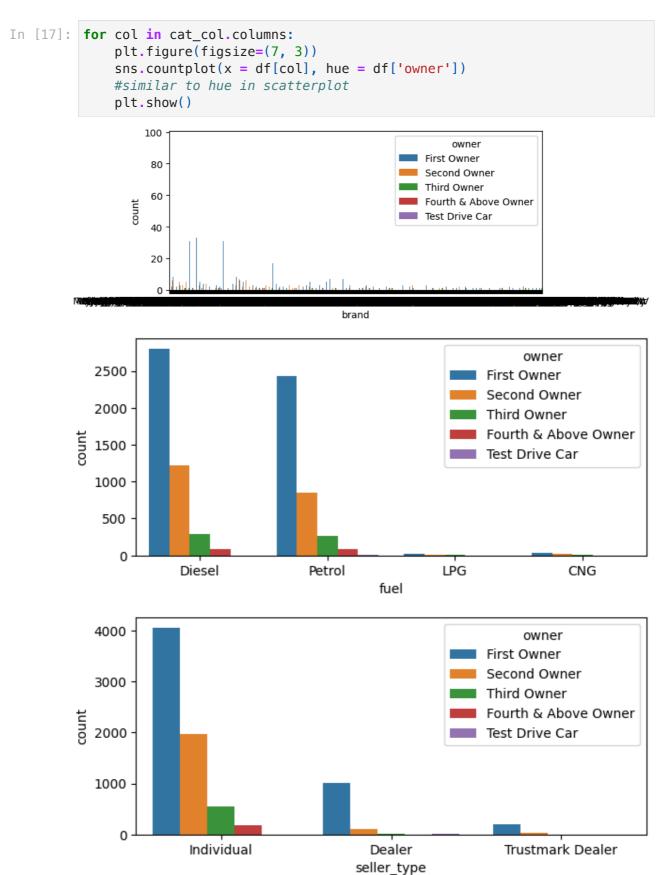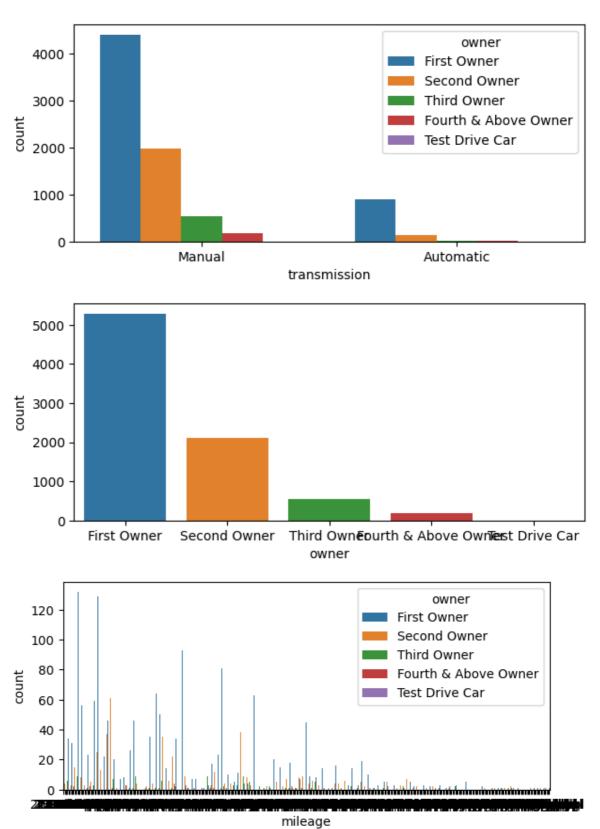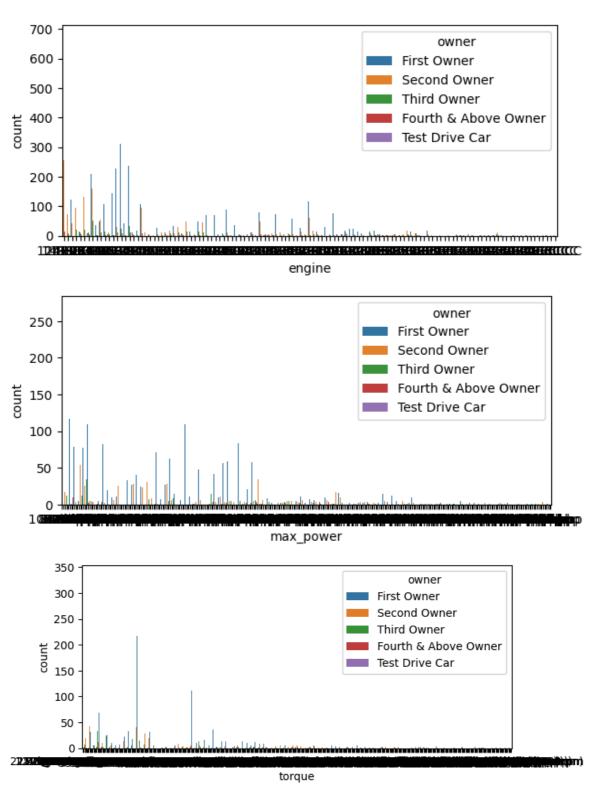## Bar Sub Plot

```
In [15]:   df.dtypes
           num_col = df.select_dtypes(include=['int64', 'float64'])
           # numcol = numcol.drop([]) #Loan_Status
           cat_col = df.select_dtypes(exclude=['int64', 'float64'])
           num_col.columns, cat_col.columns
```

```
Out[15]:   (Index(['year', 'selling_price', 'km_driven', 'seats'], dtype='object'),
            Index(['brand', 'fuel', 'seller_type', 'transmission', 'owner', 'mileag
           e',
                   'engine', 'max_power', 'torque'],
                  dtype='object'))
```

```
In [16]:   # Number of categorical columns
           num_cols = len(num_col.columns)

           # Set up the subplot grid dimensions
           rows = (num_cols + 2) // 3  #
           fig, axes = plt.subplots(rows, 3, figsize=(15, rows * 4))
           axes = axes.flatten()

           # Iterate through the categorical columns and plot countplots
           for i, col in enumerate(num_col.columns):
               sns.barplot(x = df['owner'], y = df[col],ax=axes[i])

               axes[i].set_title(f'Countplot for {col}')
               axes[i].tick_params(axis='x', rotation=45)  # Rotate x-axis labels if

           # Remove unused subplot spaces
           for j in range(i + 1, len(axes)):
               fig.delaxes(axes[j])

           # Adjust layout
           plt.tight_layout()
           plt.show()
```


Histogram Sub plot

```
In [17]:  for col in cat_col.columns:
              plt.figure(figsize=(7, 3))
              sns.countplot(x = df[col], hue = df['owner'])
              #similar to hue in scatterplot
              plt.show()
```

4. Data Preparation

```
In [18]: #Mapping Owner

         #check for unique owner
         df["owner"].unique()

         #mapping owner
         mapping = {'First Owner' : 1,
                    'Second Owner': 2,
                    'Third Owner': 3,
                    'Fourth & Above Owner': 4,
```

```
                'Test Drive Car': 5
                }

df['owner'] = df['owner'].map(mapping)

#Final Check
df["owner"].unique()
```

Out[18]: `array([1, 2, 3, 4, 5])`

In [19]:
```
#removing all rows with CNG and LPG from fuel colums
df = df[~df['fuel'].isin(['CNG', 'LPG'])]
```

In [20]:
```
#removing "kmpl" from mileague columns
df['mileage'] = df['mileage'].str.split(' ').str[0].astype(float)
```

In [21]:
```
#removing "CC" from engine columns

df['engine'] = df['engine'].str.replace('CC', '').astype(float)
```

In [22]:
```
#Removing "bhp" from Max Power columns
df['max_power'] = df['max_power'].str.replace('bhp', '').astype(float)
```

In [23]:
```
#Removing other word except the first word in brand columns
df['brand'] = df['brand'].str.split(' ').str[0]
```

In [24]:
```
#Drop "Torque" columns
df = df.drop(columns=['torque'])
```

In [25]:
```
#Removing all "Test Drive Car" from owner columns
df = df[df['owner'] != 5]
```

In [26]:
```
#Performing Log Transform
y = np.log(df['selling_price'])
```

In [27]:
```
#Final Table after cleaning
df.head()
```

Out[27]:

	brand	year	selling_price	km_driven	fuel	seller_type	transmission	owne
0	Maruti	2014	450000	145500	Diesel	Individual	Manual	
1	Skoda	2014	370000	120000	Diesel	Individual	Manual	
2	Honda	2006	158000	140000	Petrol	Individual	Manual	
3	Hyundai	2010	225000	127000	Diesel	Individual	Manual	
4	Maruti	2007	130000	120000	Petrol	Individual	Manual	

Chaning string in dataset into number

In [28]:
```
#Group different Car Brand by Region

brand_groups = {
    'Asian': [
        'Maruti', 'Tata', 'Mahindra', 'Force', 'Ambassador', 'Ashok',  #
```

```python
            'Honda', 'Toyota', 'Mitsubishi', 'Nissan', 'Lexus', 'Isuzu', 'Dat
            'Hyundai', 'Daewoo', 'Kia'  # South Korea
        ],
        'European': [
            'Renault', 'Peugeot',  # France
            'Skoda',  # Czech Republic
            'Fiat',  # Italy
            'Mercedes-Benz', 'Audi', 'Volkswagen', 'BMW', 'Opel',  # Germany
            'Jaguar', 'Land', 'MG',  # UK
            'Volvo'  # Sweden
        ],
        'USA': [
            'Ford', 'Chevrolet', 'Jeep'  # USA
        ]
    }

    # Map each brand to its own group
    def map_group(brand):
        for group, brands in brand_groups.items():
            if brand in brands:
                return group
        return 'Unknown'

    df['group'] = df['brand'].apply(map_group)

    #create one hot encoding
    df['brand_Asian'] = (df['group'] == 'Asian').astype(int)
    df['brand_European'] = (df['group'] == 'European').astype(int)
    df['brand_USA'] = (df['group'] == 'USA').astype(int)

    #drop group columns
    df = df.drop(columns=['group'])

    #drop brand columns
    df = df.drop(columns=['brand'])
```
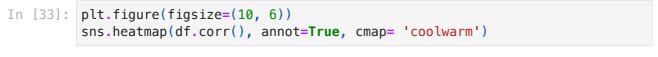
In [29]:
```python
from sklearn.preprocessing import LabelEncoder

#do label encoding for fuel
le = LabelEncoder()
df['fuel'] = le.fit_transform(df['fuel'])
```

In [30]:
```python
#do label encoding for transmission
le = LabelEncoder()
df['transmission'] = le.fit_transform(df['transmission'])
```

In [31]:
```python
# Create one-hot encoding for 'seller_type'
df['seller_type_Individual'] = (df['seller_type'] == 'Individual').astype
df['seller_type_Dealer'] = (df['seller_type'] == 'Dealer').astype(int)
df['seller_type_Trustmark_Dealer'] = (df['seller_type'] == 'Trustmark Dea

# Drop the original 'seller_type' column if it's no longer needed
df = df.drop(columns=['seller_type'])
```

In [32]:
```python
display(df)
```

	year	selling_price	km_driven	fuel	transmission	owner	mileage	engine	
0	2014	450000	145500	0	1	1	23.40	1248.0	
1	2014	370000	120000	0	1	2	21.14	1498.0	
2	2006	158000	140000	1	1	3	17.70	1497.0	
3	2010	225000	127000	0	1	1	23.00	1396.0	
4	2007	130000	120000	1	1	1	16.10	1298.0	
...	...	...	...	...	...	...	...	...	
8123	2013	320000	110000	1	1	1	18.50	1197.0	
8124	2007	135000	119000	0	1	4	16.80	1493.0	
8125	2009	382000	120000	0	1	1	19.30	1248.0	
8126	2013	290000	25000	0	1	1	23.57	1396.0	
8127	2013	290000	25000	0	1	1	23.57	1396.0	

8028 rows × 16 columns

5. Correlation (Heat Map)

```
In [33]:   plt.figure(figsize=(10, 6))
           sns.heatmap(df.corr(), annot=True, cmap= 'coolwarm')
```

Out[33]:   <Axes: >


```
In [34]:   #Dropping seat columns because there are not much effect to it according
```

```
df = df.drop('seats', axis='columns')
```

6. Processing

Null Value

In [35]: `#check the missing value in df data set`
`df.isna().sum()`

Out[35]:
```
year                          0
selling_price                 0
km_driven                     0
fuel                          0
transmission                  0
owner                         0
mileage                     214
engine                      214
max_power                   208
brand_Asian                   0
brand_European                0
brand_USA                     0
seller_type_Individual        0
seller_type_Dealer            0
seller_type_Trustmark_Dealer  0
dtype: int64
```
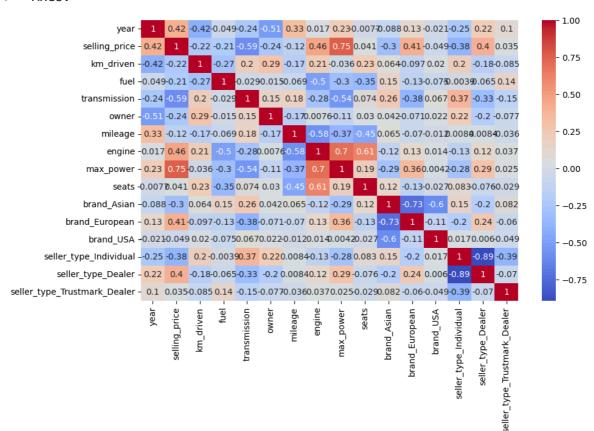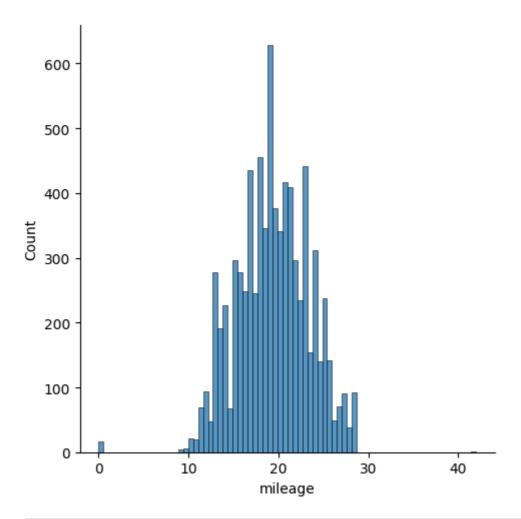
In [36]: `#Check Mileage Missing Values`
`sns.displot(df.mileage)`

Out[36]: `<seaborn.axisgrid.FacetGrid at 0x13e944ee0>`

In [37]: `#Fill out mileage with mean`

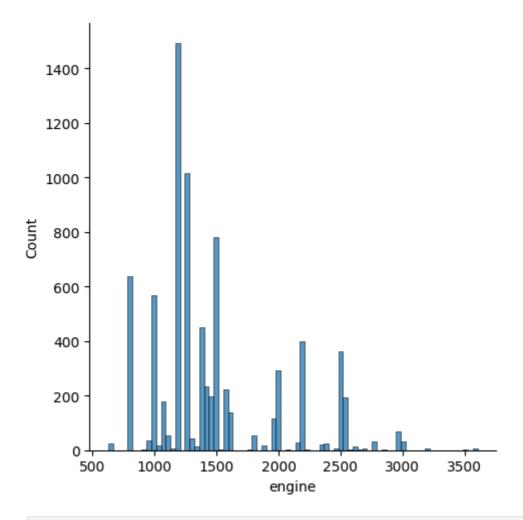`df['mileage'].fillna(df['mileage'].mean(), inplace=True)`

```
/var/folders/55/sf5l4l957zlg5xqxmtw1xg9c0000gn/T/ipykernel_63198/208828496
4.py:3: FutureWarning: A value is trying to be set on a copy of a DataFram
e or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never wor
k because the intermediate object on which we are setting values always be
haves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.


  df['mileage'].fillna(df['mileage'].mean(), inplace=True)
```

In [38]: `#Check Engine Missing Values`
`sns.displot(df.engine)`

Out[38]: `<seaborn.axisgrid.FacetGrid at 0x15f58d610>`

In [39]: ```python
#Fill out engine with median

df['engine'].fillna(df['engine'].median(), inplace=True)
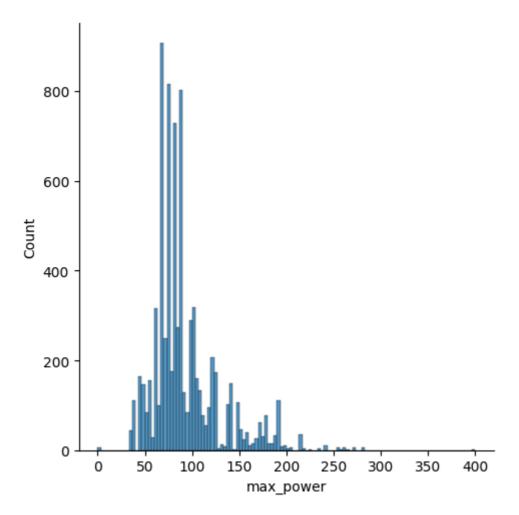```

```
/var/folders/55/sf5l4l957zlg5xqxmtw1xg9c0000gn/T/ipykernel_63198/299018232
9.py:3: FutureWarning: A value is trying to be set on a copy of a DataFram
e or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never wor
k because the intermediate object on which we are setting values always be
haves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.

  df['engine'].fillna(df['engine'].median(), inplace=True)
```

In [40]: ```python
#Check Max Power Missing Values
sns.displot(df.max_power)
```

Out[40]:  <seaborn.axisgrid.FacetGrid at 0x1691bfb80>

```
In [41]:  #Fill out max_power with median

          df['max_power'].fillna(df['max_power'].median(), inplace=True)
```

```
/var/folders/55/sf5l4l957zlg5xqxmtw1xg9c0000gn/T/ipykernel_63198/177013293
8.py:3: FutureWarning: A value is trying to be set on a copy of a DataFram
e or Series through chained assignment using an inplace method.
The behavior will change in pandas 3.0. This inplace method will never wor
k because the intermediate object on which we are setting values always be
haves as a copy.

For example, when doing 'df[col].method(value, inplace=True)', try using
'df.method({col: value}, inplace=True)' or df[col] = df[col].method(value)
instead, to perform the operation inplace on the original object.

  df['max_power'].fillna(df['max_power'].median(), inplace=True)
```

7. Modeling

```
In [42]:  #experiment tracking
          import mlflow
          import os
          # This the dockerized method.
          # We build two docker containers, one for python/jupyter and another for
          # The url `mlflow` is resolved into another container within the same com
          mlflow.set_tracking_uri("http://localhost:5100")
          # In the dockerized way, the user who runs this code will be `root`.
          # The MLflow will also log the run user_id as `root`.
```

```
# To change that, we need to set this environ["LOGNAME"] to your name.
os.environ["LOGNAME"] = "jirapon"
# mlflow.create_experiment(name="chaky-diabetes-example")  #create if you
mlflow.set_experiment(experiment_name="jirapon-regularization-example")
```

Out[42]: <Experiment: artifact_location='mlflow-artifacts:/895208492508162401', c
         reation_time=1738146927874, experiment_id='895208492508162401', last_upd
         ate_time=1738146927874, lifecycle_stage='active', name='jirapon-regulari
         zation-example', tags={}>

In [43]:
```python
#Identify x
X = df[['year','km_driven','mileage','brand_USA','brand_European','brand_

#Identify y
Y = df['selling_price']
```

In [44]:
```python
from sklearn.model_selection import train_test_split
from sklearn.impute import SimpleImputer
from sklearn.preprocessing import StandardScaler
import numpy as np

# 1️⃣ Train-Test Split (Correct)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.2

# 2️⃣ Check for Missing Values BEFORE Processing
print("Missing in X_train:", np.isnan(X_train).sum())
print("Missing in X_test:", np.isnan(X_test).sum())

# 3️⃣ Handle Missing Values (Imputation)
imputer = SimpleImputer(strategy='mean')
X_train = imputer.fit_transform(X_train)
X_test = imputer.transform(X_test)

# 4️⃣ Check for Missing Values AFTER Imputation
print("After Imputation (Train):", np.isnan(X_train).sum())
print("After Imputation (Test):", np.isnan(X_test).sum())

# 5️⃣ Standardize Features
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)

# # 6️⃣ Add Intercept Term (Optional for Linear Models)
X_train = np.concatenate((np.ones((X_train.shape[0], 1)), X_train), axis=
X_test = np.concatenate((np.ones((X_test.shape[0], 1)), X_test), axis=1)

# ✅ Check Final Shapes
assert X_train.ndim == 2
assert X_test.ndim == 2
assert y_train.ndim == 1
assert y_test.ndim == 1

print("Final Training features shape:", X_train.shape)
print("Final Testing features shape:", X_test.shape)
```

```
Missing in X_train: year                  0
km_driven          0
mileage            0
brand_USA          0
brand_European     0
brand_Asian        0
dtype: int64
Missing in X_test: year                   0
km_driven          0
mileage            0
brand_USA          0
brand_European     0
brand_Asian        0
dtype: int64
After Imputation (Train): 0
After Imputation (Test): 0
Final Training features shape: (6422, 7)
Final Testing features shape: (1606, 7)
```

In [45]:
```python
from sklearn.model_selection import KFold

class LinearRegression(object):

    #in this class, we add cross validation as well for some spicy code..
    kfold = KFold(n_splits=3)

    def __init__(self, regularization, lr=0.001, method='batch', num_epoc
        self.lr         = lr
        self.num_epochs = num_epochs
        self.batch_size = batch_size
        self.method     = method
        self.cv         = cv
        self.regularization = regularization

    def mse(self, ytrue, ypred):
        return ((ypred - ytrue) ** 2).sum() / ytrue.shape[0]

    def r2(self, y_true, y_pred):
        ss_total = np.sum((y_true - np.mean(y_true)) ** 2)
        ss_residual = np.sum((y_true - y_pred) ** 2)
        return 1 - (ss_residual / ss_total)

    def fit(self, X_train, y_train):

        #create a list of kfold scores
        self.kfold_scores = list()

        #reset val loss
        self.val_loss_old = np.inf

        #kfold.split in the sklearn.....
        #5 splits
        for fold, (train_idx, val_idx) in enumerate(self.cv.split(X_train

            X_cross_train = X_train[train_idx]
            y_cross_train = y_train[train_idx]
            X_cross_val   = X_train[val_idx]
            y_cross_val   = y_train[val_idx]

            self.theta = np.zeros(X_cross_train.shape[1])
```

```python
                #define X_cross_train as only a subset of the data
                #how big is this subset?  => mini-batch size ==> 50

                #one epoch will exhaust the WHOLE training set
                with mlflow.start_run(run_name=f"Fold-{fold}", nested=True):

                    params = {"method": self.method, "lr": self.lr, "reg": ty
                    mlflow.log_params(params=params)

                    for epoch in range(self.num_epochs):

                        #with replacement or no replacement
                        #with replacement means just randomize
                        #with no replacement means 0:50, 51:100, 101:150, ...
                        #shuffle your index
                        perm = np.random.permutation(X_cross_train.shape[0])

                        X_cross_train = X_cross_train[perm]
                        y_cross_train = y_cross_train[perm]

                        if self.method == 'sto':
                            for batch_idx in range(X_cross_train.shape[0]):
                                X_method_train = X_cross_train[batch_idx].res
                                y_method_train = y_cross_train[batch_idx]
                                train_loss = self._train(X_method_train, y_me
                        elif self.method == 'mini':
                            for batch_idx in range(0, X_cross_train.shape[0],
                                #batch_idx = 0, 50, 100, 150
                                X_method_train = X_cross_train[batch_idx:batc
                                y_method_train = y_cross_train[batch_idx:batc
                                train_loss = self._train(X_method_train, y_me
                        else:
                            X_method_train = X_cross_train
                            y_method_train = y_cross_train
                            train_loss = self._train(X_method_train, y_method

                        mlflow.log_metric(key="train_loss", value=train_loss,

                        yhat_val = self.predict(X_cross_val)
                        val_loss_new = self.mse(y_cross_val, yhat_val)
                        mlflow.log_metric(key="val_loss", value=val_loss_new,

                        #early stopping
                        if np.allclose(val_loss_new, self.val_loss_old):
                            break
                        self.val_loss_old = val_loss_new

                    self.kfold_scores.append(val_loss_new)
                    print(f"Fold {fold}: {val_loss_new}")


    def _train(self, X, y):
        yhat = self.predict(X)
        m    = X.shape[0]
        grad = (1/m) * X.T @(yhat - y) + self.regularization.derivation(s
        self.theta = self.theta - self.lr * grad
        return self.mse(y, yhat)

    def predict(self, X):
```

```python
        return X @ self.theta  #===>(m, n) @ (n, )

    def _coef(self):
        return self.theta[1:]

    def _bias(self):
        return self.theta[0]
```

In [46]:
```python
class LassoPenalty:

    def __init__(self, l):
        self.l = l # lambda value

    def __call__(self, theta): #__call__ allows us to call class as metho
        return self.l * np.sum(np.abs(theta))

    def derivation(self, theta):
        return self.l * np.sign(theta)

class RidgePenalty:

    def __init__(self, l):
        self.l = l

    def __call__(self, theta): #__call__ allows us to call class as metho
        return self.l * np.sum(np.square(theta))

    def derivation(self, theta):
        return self.l * 2 * theta

class ElasticPenalty:

    def __init__(self, l = 0.1, l_ratio = 0.5):
        self.l = l
        self.l_ratio = l_ratio

    def __call__(self, theta):  #__call__ allows us to call class as meth
        l1_contribution = self.l_ratio * self.l * np.sum(np.abs(theta))
        l2_contribution = (1 - self.l_ratio) * self.l * 0.5 * np.sum(np.s
        return (l1_contribution + l2_contribution)

    def derivation(self, theta):
        l1_derivation = self.l * self.l_ratio * np.sign(theta)
        l2_derivation = self.l * (1 - self.l_ratio) * theta
        return (l1_derivation + l2_derivation)

class Lasso(LinearRegression):

    def __init__(self, method, lr, l):
        self.regularization = LassoPenalty(l)
        super().__init__(self.regularization, lr, method)

class Ridge(LinearRegression):

    def __init__(self, method, lr, l):
        self.regularization = RidgePenalty(l)
        super().__init__(self.regularization, lr, method)

class ElasticNet(LinearRegression):
```

```python
    def __init__(self, method, lr, l, l_ratio=0.5):
        self.regularization = ElasticPenalty(l, l_ratio)
        super().__init__(self.regularization, lr, method)
```

In [47]:
```python
# pseudocode for xavier weight initialization

m = 10   #number of sample

lower , upper = -(1.0 / np.sqrt(m)), (1.0 / np.sqrt(m)) # calculate the r

print (lower , upper ) # summarize the range

numbers = np.random.rand(1000) # generate random numbers, randomly pick w
scaled = lower + numbers * ( upper - lower )
print ( scaled )
```

```
      -0.31622776601683794 0.31622776601683794
      [-0.30526361   0.23696335  -0.11167621  -0.16978311  -0.14721462   0.09440965
        0.15207459   0.28118693   0.22182802   0.28705366  -0.14789815  -0.19563491
        0.19472345   0.19451799   0.17393911  -0.02864771  -0.05023523   0.15200472
       -0.04799737  -0.22867846  -0.07421324   0.07187834  -0.14073226   0.14418826
       -0.28905244   0.03167589   0.27851342  -0.12881583   0.06085415   0.19334557
        0.07988131  -0.07657498   0.09737259  -0.12629414   0.10442272  -0.23037842
        0.08337846   0.257066     0.11478916  -0.1357232   -0.15132083   0.27544306
       -0.27632944  -0.13196758  -0.10790351  -0.22829677  -0.01259211  -0.19303911
        0.11291871  -0.10254673  -0.06330785  -0.14560705  -0.14497875   0.03927616
        0.11968607  -0.26881347   0.17155408  -0.02238128   0.05329375   0.31392097
       -0.02855205   0.17189518   0.15605541  -0.05633945  -0.26589909   0.19782715
       -0.15747662  -0.10048097  -0.01055681   0.22595039   0.15770432   0.22155273
        0.00266941  -0.09729019   0.26983336  -0.02327404  -0.31237282  -0.00225532
       -0.22843515  -0.15498461  -0.17886426  -0.17089831   0.12445377  -0.05788791
        0.1071176   -0.12263962   0.28331552   0.05096804  -0.16224823  -0.18366952
        0.1547437   -0.28202148  -0.05242717  -0.26282111  -0.00542025   0.1323593
        0.04713426   0.01022837   0.12816064   0.16054607   0.29722652  -0.04226128
        0.06124726  -0.05548872   0.25904565   0.25212087  -0.00525815  -0.05907963
       -0.09090769  -0.29439045  -0.02874465  -0.31158168   0.20986191   0.2893746
       -0.23520818   0.10672158  -0.06683519   0.29746841  -0.24977069  -0.07472858
       -0.02910324  -0.06740566   0.16314777  -0.23721044   0.15307239   0.1939834
        0.06946594  -0.21642893   0.21453611  -0.2377556   -0.22665054  -0.22641307
       -0.28599423  -0.30918784  -0.14100283   0.17085396  -0.03407296  -0.01042039
       -0.17214377   0.06256984   0.18252959  -0.12328271   0.21093094   0.2922656
       -0.28960305   0.31297893  -0.1309528   -0.1480388    0.17746284  -0.10795421
        0.14291477  -0.22653957  -0.14336906  -0.09066471  -0.2948896   -0.30091853
       -0.09568222  -0.03671848   0.28151417  -0.107087     0.31499919  -0.00684104
        0.09679328   0.05795135   0.28103234  -0.23129821   0.19460324   0.07558073
        0.10270885  -0.03712942   0.03075265   0.22968152  -0.19371411   0.20697733
       -0.10436156  -0.22805303  -0.06431754  -0.29763118   0.14022907   0.25315869
        0.23765146  -0.03938214  -0.11206205   0.11536901   0.2363513   -0.19900477
       -0.22970703   0.11991257   0.24925438   0.09150492   0.22814648   0.017201
       -0.18610896  -0.00382027   0.0377389    0.24256577  -0.20970689  -0.14583533
        0.02031139   0.13960114   0.14206727   0.16385341  -0.25160279   0.17766125
       -0.01557275   0.11187394  -0.2108611    0.20010226  -0.07048113   0.27605357
        0.2848748   -0.20505808   0.10572613  -0.2994286    0.1349501    0.23825603
        0.10676779  -0.28323128   0.14375934   0.30433139   0.23519405   0.03848064
       -0.13363417  -0.14904236  -0.02846427  -0.19687765   0.1683038    0.21677287
       -0.02793981   0.10019856  -0.18565893   0.05285875  -0.21713169  -0.29968059
       -0.05614944  -0.22281054  -0.01552444   0.19872615   0.22587441   0.06880124
       -0.03593199  -0.20161696  -0.07873093  -0.20756031  -0.11960641  -0.09585825
       -0.15630915   0.13323445   0.09217723   0.02102341   0.20352639  -0.22084064
       -0.14174466  -0.13544838   0.03219739   0.11456541  -0.17841734   0.13324645
       -0.17973786   0.18384888  -0.21837751  -0.21470387   0.09366017  -0.21868768
        0.00207615   0.13848334  -0.13114637  -0.16028351  -0.26019679  -0.30484117
        0.07611785   0.15027185   0.11754608   0.21036059   0.00831386   0.24522143
       -0.01304941  -0.23052491  -0.07415858   0.08191302   0.21958569  -0.0990173
       -0.01112592   0.30275447  -0.10107333  -0.17603792   0.21633739   0.06990086
       -0.16531659  -0.2122036   -0.21925369   0.25322144   0.29975494  -0.22236861
       -0.22228173   0.252491    -0.10516099   0.1857491   -0.0704709   -0.25448185
        0.0787456   -0.21514822  -0.2417663    0.10736724   0.1760201    0.04585634
       -0.10626417   0.16266621   0.02964934   0.25687099   0.07322203  -0.22663828
        0.22697719   0.09910516  -0.13096069  -0.21768166  -0.13650899   0.10589147
       -0.25884828   0.12939561   0.21476429  -0.17541343   0.17294294   0.29475497
       -0.22357005   0.29332194  -0.07572154  -0.25764469   0.26285641  -0.11351681
        0.04587385  -0.02507656  -0.04926701  -0.16992093  -0.03688442   0.07227445
       -0.1565822    0.05279736   0.17037164  -0.18268859   0.02024231   0.08147884
        0.27905143  -0.18749269  -0.3120285   -0.01905052  -0.28184223   0.1197633
        0.29038268   0.24772336  -0.00927138  -0.08612198   0.23335765  -0.30394866
```

```
 0.07895334  -0.28629281  -0.18738415  -0.26663811  -0.310055    -0.13689224
-0.00587544  -0.19533275  -0.30329567   0.25695119   0.15204903  -0.14956895
 0.29910522  -0.05833995  -0.17398595   0.16443541  -0.00054485  -0.06142138
 0.06610144   0.29083035  -0.26823847  -0.09151295  -0.28183594   0.00495204
-0.16575777   0.27947324  -0.25351406   0.28582846   0.14244911   0.17563163
 0.07661677   0.27371765  -0.11710324   0.1378498   -0.13894741   0.0047858
-0.18234072   0.25258996   0.13246482  -0.12651637  -0.26759648   0.21875851
-0.26425869   0.01451746   0.24673032  -0.21207244   0.19218036  -0.00223104
-0.05941911   0.28709854  -0.31088354   0.21382379  -0.20282765  -0.05655161
 0.21066144   0.17053966   0.28953703   0.05687613  -0.21516681   0.09144877
-0.03211441  -0.06260372  -0.30672357   0.30991152   0.01084784  -0.26938042
 0.06860751   0.0493724   -0.24448978  -0.14833813  -0.06321198  -0.19030181
-0.11846024  -0.05408401  -0.23956993  -0.08070356  -0.04472182  -0.03633052
 0.19547751   0.2827132   -0.31013244   0.1556373   -0.14427589   0.30556637
-0.03405154   0.22900254  -0.10576427  -0.10834118   0.06754512  -0.22436082
 0.21897153   0.20182252  -0.00581546   0.2155647    0.0344175   -0.14851434
-0.02083197  -0.02891246  -0.26638198   0.17681564  -0.09466582  -0.30303761
-0.17971056  -0.16265644  -0.02285656   0.31182413   0.26581518  -0.02330646
-0.247099    -0.03518573  -0.08650948   0.01022064   0.16892582  -0.29775512
-0.21429707  -0.20403063   0.16133954  -0.27538497   0.25641505  -0.1628214
 0.23137616   0.28102625   0.01453678   0.01838095  -0.254176     0.11953214
-0.02531029   0.24473226   0.03940731   0.02610796  -0.19305761  -0.01627432
-0.18122991   0.01177025   0.06502364  -0.10443803  -0.28693756  -0.01405115
 0.06697002   0.19453604  -0.03064614  -0.1431116   -0.20105251   0.13498448
 0.04714921   0.11650263  -0.13348526   0.07547321   0.10443017  -0.25992555
 0.18779755   0.2354864   -0.0160414   -0.29400138   0.28676347   0.25123074
-0.22797094  -0.17683716   0.01538766   0.24769044   0.1243869    0.1172752
-0.25232351  -0.07125078  -0.30662234   0.09151221  -0.11132081   0.21337444
-0.11858764   0.16388912  -0.25830811   0.21619408  -0.27407203   0.31521685
 0.16720877  -0.06881665  -0.26682457   0.02615736   0.30689796   0.29122138
-0.27922847   0.19902503  -0.02316911  -0.02772532   0.3005872    0.09325854
-0.06191181   0.17206722  -0.00581241   0.03556602   0.08033621   0.15043815
 0.02167905   0.1643617    0.26550532   0.0344719    0.13035624   0.05639858
-0.18983428   0.28964435   0.28573286  -0.26618381  -0.1255935   -0.0139719
-0.09638143  -0.07138703  -0.0424775   -0.13082981   0.19511148   0.14766997
 0.21607037  -0.04948061   0.15373207   0.2635118    0.01502288  -0.29294347
 0.19195048   0.31187898  -0.21800807   0.08348057  -0.25326945  -0.31401495
 0.03282252  -0.21579575   0.12343577   0.26421524   0.08561152   0.21172545
-0.14613938   0.03040845  -0.18947641   0.12055023   0.15656406  -0.19234698
 0.00964701  -0.02418301   0.29263359  -0.15912808  -0.24492462   0.03743372
 0.02907502   0.22108963   0.2264588   -0.23161461  -0.26614837   0.18655304
-0.28580888  -0.19907512   0.20380051  -0.1808568   -0.28296855   0.08197658
-0.0383892   -0.05304089   0.25970793   0.22003657  -0.18568935   0.28765493
-0.30925978   0.28432441  -0.13705123  -0.08428029   0.27408194  -0.18357321
 0.09929451  -0.1269117    0.05128787   0.08881625  -0.14164901   0.14667514
 0.00838373  -0.06003305   0.14758426  -0.15604642  -0.26502255   0.30874014
-0.12272832  -0.10705946  -0.14890358  -0.30425621  -0.14023056   0.10027838
 0.16934501   0.26221353  -0.22741294   0.10982636   0.22824365  -0.08849645
-0.24487078   0.00073407  -0.04606039   0.10096127  -0.13979269   0.18781849
 0.29845033  -0.22381502   0.07505626  -0.18961066  -0.28426972  -0.057725
 0.00576665   0.09587182   0.08772785   0.20139574   0.22543808  -0.27820055
-0.02142715   0.2741115    0.27890045   0.15839499  -0.23616723   0.23104692
-0.1257786    0.19970525   0.00617181   0.124684     0.02475299  -0.20571081
 0.22277488  -0.14758259   0.15211688  -0.2954282   -0.13953606   0.23610639
 0.23813061   0.19049388  -0.05944787  -0.19189522  -0.24927097  -0.07198792
 0.10815039  -0.01524825   0.20276182   0.20816785  -0.24754622  -0.20753955
 0.29281792   0.29777283   0.20794691  -0.11037916   0.16170041   0.24518474
-0.0419688    0.13050946  -0.17975589   0.18894019   0.05702081  -0.18977792
 0.07584925  -0.1440183   -0.16031372   0.01841077  -0.06980929  -0.13513847
-0.11953559  -0.0373962    0.163425    -0.08010496  -0.21506792   0.29231061
```

```
 0.08105779 -0.24363915 -0.2690444   0.06146122  0.15748993 -0.13683055
 0.0015508  -0.07780046  0.24148371 -0.25094271 -0.05421891 -0.26316544
 0.05723126 -0.23317613 -0.03912267 -0.07459815 -0.18327502 -0.02073919
 0.30566565  0.22167358 -0.01459203 -0.02266754 -0.17637223 -0.0617435
-0.16073236  0.17902577 -0.18203274 -0.3066854  -0.19187406 -0.17414745
-0.02503859 -0.200198    0.17078832  0.00097987 -0.29772126  0.15803549
 0.07904614 -0.08516456 -0.2848643  -0.05778537  0.16333307  0.01860651
 0.17237875  0.12321543 -0.07504213 -0.14925463  0.12276654  0.07732488
-0.30536013  0.10106205 -0.22759019  0.14857854 -0.16011     0.18909474
 0.12128841 -0.21150088  0.1844511  -0.23907133  0.18417539  0.14729207
-0.00701949  0.24525198  0.26245826  0.15655938  0.28962905 -0.24213803
 0.14292539  0.13740764  0.05949639 -0.07836744  0.24456133 -0.01049312
 0.05796485 -0.11485295 -0.21041149  0.13813242  0.17753762  0.10097075
 0.15840749  0.07681422 -0.17385424  0.1323132   0.21053243 -0.0631559
-0.03676305 -0.02017209  0.27867491 -0.1777627  -0.16539591  0.17111936
-0.28540225 -0.29294293 -0.15726824  0.23182836 -0.05539827 -0.11446619
 0.24285514 -0.01129879  0.10383202  0.06978576  0.13860659 -0.1091517
-0.27702174  0.27002927 -0.19823613  0.25195553  0.00708284  0.23415019
 0.13371033  0.23093043  0.13300549  0.12578499  0.01381718  0.30886806
 0.11700089  0.12614189 -0.04818033  0.07958372  0.24660917 -0.31609184
 0.16219644 -0.23082025 -0.09613053 -0.17184009  0.19252696  0.11475134
-0.30453855 -0.0012476   0.02032649  0.2370174   0.09751012 -0.11482776
-0.10507261 -0.22799882  0.03134815 -0.09141562 -0.1312507  -0.0443959
 0.05379951  0.26502042 -0.24197706  0.00864692  0.27772839  0.11648313
-0.01446012  0.12428944 -0.25566276 -0.25285502  0.07584319 -0.22887261
-0.15620261  0.26246466 -0.16852296  0.16455477  0.26293165  0.31291412
-0.06301504  0.30946222 -0.01177951  0.07398936  0.30991882 -0.06836465
 0.07005057 -0.10595262 -0.17104268 -0.30501531  0.14452172 -0.18733882
 0.05008941  0.17186487 -0.11247572  0.18587976  0.28979285  0.02046702
 0.18408072  0.19132884 -0.02560656  0.12229224  0.00240349  0.22166516
-0.1132584   0.30163427  0.08024214 -0.0818167  -0.15959616  0.08810119
 0.27182454 -0.06654459  0.05747203  0.04705192  0.21189887  0.1094343
-0.22667587 -0.19931661  0.03479979 -0.21221993 -0.15548804 -0.11925141
-0.167994   -0.16471761 -0.30584549  0.27082482 -0.09163824  0.01320289
 0.07070375 -0.25754888  0.17469832  0.21831011  0.23451645 -0.27644547
-0.08003963 -0.26869082  0.02915577  0.08455558  0.08224766 -0.14260977
-0.14733216 -0.30066099  0.16402742  0.18838972  0.23967113  0.30508735
-0.1914091   0.10670042 -0.06277692  0.03179165  0.25715469 -0.19238898
-0.24410722 -0.0210967  -0.01465328  0.01916342  0.23173182 -0.21841502
 0.15792627  0.19533601  0.04600424  0.06691994 -0.16694984 -0.17459964
-0.10896814  0.25378226 -0.14681115 -0.1247816   0.20739276 -0.30993497
 0.16677817  0.0243389   0.02890308 -0.05166023 -0.06369382  0.08515007
-0.13487143 -0.24915971 -0.09929008  0.19882964 -0.14874939  0.114203
 0.2871544  -0.29863567 -0.17268118 -0.11623703  0.09302503 -0.24679158
 0.30433447 -0.25002034 -0.28728548 -0.29605263 -0.11817081  0.09224625
 0.10424968 -0.30996674 -0.18761092 -0.00879559 -0.29392906  0.20625317
-0.29378898 -0.09509514 -0.25736343 -0.24542576 -0.1793749  -0.10905292
 0.17907511  0.20134559 -0.22542588 -0.29204758]
```

```
In [48]:  # pseudocode for momentum
          alpha = 0.01        # Learning rate
          grad = 0.1          # Example gradient value
          momentum = 0.9      # Momentum coefficient
          theta = 1.0         # Initial parameter value
          prev_step = 0.0     # Initial previous step


          def update():
              step = alpha * grad
              theta = theta - step + momentum * prev_step
              prev_step = step
```

8. Experiment

```
In [49]:  #helper function for looping classnames
          import sys

          def str_to_class(classname):
              return getattr(sys.modules[__name__], classname)
```

```
In [50]:  import mlflow
          from sklearn.linear_model import Ridge, Lasso, ElasticNet
          from sklearn.metrics import mean_squared_error, r2_score
          import numpy as np

          # Ensure y_train is a NumPy array to avoid indexing issues
          y_train = np.array(y_train)

          # Mapping for model names to classes
          def str_to_class(classname):
              model_classes = {
                  "Ridge": Ridge,
                  "Lasso": Lasso,
                  "ElasticNet": ElasticNet
              }
              return model_classes.get(classname, None)

          # Regularization models to run
          regs = ["Ridge", "Lasso", "ElasticNet"]

          for reg in regs:
              # Correct parameters for Ridge, Lasso, ElasticNet
              params = {"alpha": 0.1}  # Regularization strength

              # Start MLflow run
              mlflow.start_run(run_name=f"alpha-{params['alpha']}-reg-{reg}", neste
              print("=" * 5, reg, "=" * 5)

              # Initialize and train the model
              type_of_regression = str_to_class(reg)
              if type_of_regression is None:
                  print(f"Unknown regression type: {reg}")
                  continue

              model = type_of_regression(**params)
              model.fit(X_train, y_train)

              # Predictions
```

```python
    yhat = model.predict(X_test)

    # Metrics
    mse = mean_squared_error(y_test, yhat)
    r2_value = r2_score(y_test, yhat)

    # Log metrics
    print("Test MSE:", mse)
    mlflow.log_metric(key="test_mse", value=mse)

    print("Test R²:", r2_value)
    mlflow.log_metric(key="test_r2", value=r2_value)  # Corrected the key

    # Log the model
    signature = mlflow.models.infer_signature(X_train, model.predict(X_tr
    mlflow.sklearn.log_model(model, artifact_path='model', signature=sign

    # End MLflow run
    mlflow.end_run()
```

```
===== Ridge =====
Test MSE: 0.2424121022935838
Test R²: 0.6439518803198924
🧍 View run alpha-0.1-reg-Ridge at: http://localhost:5100/#/experiments/89
5208492508162401/runs/29b4d24b27bc4863a4e740c34027cbc3
🖊 View experiment at: http://localhost:5100/#/experiments/895208492508162
401
===== Lasso =====
Test MSE: 0.29781115922143064
Test R²: 0.562583294079414
🧍 View run alpha-0.1-reg-Lasso at: http://localhost:5100/#/experiments/89
5208492508162401/runs/dde5e2dbdd7349f680f23e671f4aa419
🖊 View experiment at: http://localhost:5100/#/experiments/895208492508162
401
===== ElasticNet =====
Test MSE: 0.27066035933544497
Test R²: 0.6024616300030408
🧍 View run alpha-0.1-reg-ElasticNet at: http://localhost:5100/#/experimen
ts/895208492508162401/runs/1e75554ef5a946cda4e50578b9b1244d
🖊 View experiment at: http://localhost:5100/#/experiments/895208492508162
401
```

Cross Validation for Linear Regression, Ridge, Lasso, Polynomial Regression

```python
In [51]: from sklearn.linear_model import Ridge, Lasso, LinearRegression
         from sklearn.preprocessing import PolynomialFeatures
         from sklearn.pipeline import make_pipeline
         from sklearn.model_selection import cross_val_score
         import numpy as np

         # Define models
         models = {
             "Linear Regression": LinearRegression(),
             "Ridge": Ridge(alpha=0.1),
             "Lasso": Lasso(alpha=0.1),
             "Polynomial Regression": make_pipeline(PolynomialFeatures(degree=2),
         }

         # MLflow Experiment
```

```python
mlflow.set_experiment("Model_Comparison")

for name, model in models.items():
    with mlflow.start_run(run_name=name, nested=True):
        mse_scores = cross_val_score(model, X_train, y_train, scoring="ne
        mse = -np.mean(mse_scores)

        r2_scores = cross_val_score(model, X_train, y_train, scoring="r2"
        r2 = np.mean(r2_scores)

        mlflow.log_metric("Cross-Validation MSE", mse)
        mlflow.log_metric("Cross-Validation R²", r2)

        # ✅ Print Results
        print(f"📊 {name}:")
        print(f"  - Cross-Validation MSE: {mse:.4f}")
        print(f"  - Cross-Validation R²:  {r2:.4f}\n")

        mlflow.end_run()
```

```
📊 Linear Regression:
  - Cross-Validation MSE: 0.2757
  - Cross-Validation R²:  0.6096

🏃 View run Linear Regression at: http://localhost:5100/#/experiments/5025
57671757064443/runs/9ba5ae6788c94b37b6866e23467b6d0e
🧪 View experiment at: http://localhost:5100/#/experiments/502557671757064
443
📊 Ridge:
  - Cross-Validation MSE: 0.2757
  - Cross-Validation R²:  0.6096

🏃 View run Ridge at: http://localhost:5100/#/experiments/5025576717570644
43/runs/f52cf8a1aaa748e1a1810d4f9a438b3e
🧪 View experiment at: http://localhost:5100/#/experiments/502557671757064
443
📊 Lasso:
  - Cross-Validation MSE: 0.3119
  - Cross-Validation R²:  0.5583

🏃 View run Lasso at: http://localhost:5100/#/experiments/5025576717570644
43/runs/441c584fb02e40a5ba36b77ec1355e62
🧪 View experiment at: http://localhost:5100/#/experiments/502557671757064
443
📊 Polynomial Regression:
  - Cross-Validation MSE: 0.2566
  - Cross-Validation R²:  0.6369

🏃 View run Polynomial Regression at: http://localhost:5100/#/experiments/
502557671757064443/runs/31201b598840455eaa24994491879bc7
🧪 View experiment at: http://localhost:5100/#/experiments/502557671757064
443
```

Implementing With and Without Momentum

```python
In [52]:  from sklearn.linear_model import SGDRegressor
          from sklearn.model_selection import cross_val_score
          import mlflow
          import numpy as np
```

```python
# Define configurations for "without momentum" and "with momentum-like ef
momentum_configs = {
    "No Momentum": SGDRegressor(alpha=0.1, max_iter=1000, learning_rate='
    "With Momentum (Averaging)": SGDRegressor(alpha=0.1, max_iter=1000, l
}

# MLflow Experiment
mlflow.set_experiment("Momentum_Comparison")

for name, model in momentum_configs.items():
    with mlflow.start_run(run_name=f"SGD_{name.replace(' ', '_')}",nested
        mse_scores = cross_val_score(model, X_train, y_train, scoring="ne
        mse = -np.mean(mse_scores)

        r2_scores = cross_val_score(model, X_train, y_train, scoring="r2"
        r2 = np.mean(r2_scores)

        # Log parameters and metrics to MLflow
        mlflow.log_param("Momentum", name)
        mlflow.log_metric("MSE", mse)
        mlflow.log_metric("R²", r2)

        # ✅ Print Results
        print(f"📊 {name}:")
        print(f"  - MSE: {mse:.4f}")
        print(f"  - R²:  {r2:.4f}\n")

        mlflow.end_run()
```

📊 No Momentum:
  - MSE: 0.2904
  - R²: 0.5250

🏃 View run SGD_No_Momentum at: http://localhost:5100/#/experiments/718027
092403891502/runs/3e0ac5b25e084af0a6b6f02cd5709e5f
🖊 View experiment at: http://localhost:5100/#/experiments/718027092403891
502
📊 With Momentum (Averaging):
  - MSE: 0.2821
  - R²: 0.6004

🏃 View run SGD_With_Momentum_(Averaging) at: http://localhost:5100/#/expe
riments/718027092403891502/runs/c244e1385f8a4592bdd2ff7a3a3ba289
🖊 View experiment at: http://localhost:5100/#/experiments/718027092403891
502

Implementing Stochastic, Mini-Batch, and Batch Gradient Descent

```python
In [53]:
from sklearn.utils import shuffle
from sklearn.metrics import mean_squared_error, r2_score

# Set Cross-Validation (5 folds)
cv = KFold(n_splits=5, shuffle=True, random_state=42)

# Custom Training Function for Gradient Descent Variants
def custom_train(X_train, y_train, batch_size):
    model = SGDRegressor(max_iter=1, learning_rate='constant', eta0=0.01,
    for epoch in range(50):
        X_shuffled, y_shuffled = shuffle(X_train, y_train)
        for i in range(0, X_train.shape[0], batch_size):
```

```python
            X_batch = X_shuffled[i:i + batch_size]
            y_batch = y_shuffled[i:i + batch_size]
            model.partial_fit(X_batch, y_batch)
    return model

# Gradient Descent Methods with Corresponding Batch Sizes
methods = {
    "Stochastic": 1,                        # Stochastic Gradient Descent (
    "Mini-Batch": 64,                       # Mini-Batch Gradient Descent (
    "Batch": X_train.shape[0]               # Batch Gradient Descent (full
}

# MLflow Experiment Setup
mlflow.set_experiment("GD_Cross_Validation_Comparison")

# Run Cross-Validation for Each Method
for method_name, batch_size in methods.items():
    with mlflow.start_run(run_name=method_name,nested=True):
        mse_scores = []
        r2_scores = []

        # 5-Fold Cross-Validation Loop
        for train_index, val_index in cv.split(X_train):
            X_train_cv, X_val_cv = X_train[train_index], X_train[val_inde
            y_train_cv, y_val_cv = y_train[train_index], y_train[val_inde

            # Train the Model with the Current Batch Size
            model = custom_train(X_train_cv, y_train_cv, batch_size)

            # Predict and Evaluate on Validation Set
            y_pred = model.predict(X_val_cv)
            mse = mean_squared_error(y_val_cv, y_pred)
            r2 = r2_score(y_val_cv, y_pred)

            mse_scores.append(mse)
            r2_scores.append(r2)

        # Average MSE and R² Across 5 Folds
        avg_mse = np.mean(mse_scores)
        avg_r2 = np.mean(r2_scores)

        # Log Metrics to MLflow
        mlflow.log_param("Method", method_name)
        mlflow.log_param("Batch Size", batch_size)
        mlflow.log_metric("Cross-Validation MSE", avg_mse)
        mlflow.log_metric("Cross-Validation R²", avg_r2)

        # Print Results
        print(f"📊 {method_name} (5-Fold Cross-Validation):")
        print(f"  - Average MSE: {avg_mse:.4f}")
        print(f"  - Average R²:  {avg_r2:.4f}\n")

        mlflow.end_run()
```

📊 Stochastic (5-Fold Cross-Validation):
  - Average MSE: 0.3097
  - Average R²: 0.5600

🏃 View run Stochastic at: http://localhost:5100/#/experiments/53335888171
2357097/runs/0898416049914ecfa0f8c97367d32fb3
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
📊 Mini-Batch (5-Fold Cross-Validation):
  - Average MSE: 0.2866
  - Average R²: 0.5932

🏃 View run Mini-Batch at: http://localhost:5100/#/experiments/53335888171
2357097/runs/f8ac62841c2c43a987aeb3dae660781c
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
📊 Batch (5-Fold Cross-Validation):
  - Average MSE: 0.2942
  - Average R²: 0.5812

🏃 View run Batch at: http://localhost:5100/#/experiments/5333588817123570
97/runs/422da68c4928406989374e4e5d2616de
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Implementing Zero and Xavier Weight Initialization

```python
In [54]: def initialize_weights(shape, method="xavier"):
    if method == "zero":
        return np.zeros(shape)
    elif method == "xavier":
        limit = np.sqrt(1 / shape)
        return np.random.uniform(-limit, limit, shape)
    else:
        raise ValueError("Invalid Initialization Method")


initializations = ["zero", "xavier"]

for init in initializations:
    with mlflow.start_run(run_name=f"Init_{init}",nested=True):
        weights = initialize_weights(X_train.shape[1], method=init)

        model = SGDRegressor(max_iter=1000, eta0=0.01)
        model.fit(X_train, y_train)

        y_pred = model.predict(X_test)
        mse = np.mean((y_test - y_pred) ** 2)
        r2 = model.score(X_test, y_test)

        mlflow.log_param("Initialization", init)
        mlflow.log_metric("MSE", mse)
        mlflow.log_metric("R²", r2)

        # ✅ Print Results
        print(f"📊 {init.capitalize()} Initialization:")
        print(f"  - MSE: {mse:.4f}")
        print(f"  - R²:  {r2:.4f}\n")

        mlflow.end_run()
```

📊 Zero Initialization:
  - MSE: 0.2411
  - R²: 0.6459

🏃 View run Init_zero at: http://localhost:5100/#/experiments/533358881712357097/runs/bcab3975ae5544fea3c1c724263ccdd2
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357097
📊 Xavier Initialization:
  - MSE: 0.2455
  - R²: 0.6394

🏃 View run Init_xavier at: http://localhost:5100/#/experiments/533358881712357097/runs/de8f2c10f94e48ec9cf49c939f2ce5d6
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Testing with Learning Rates 0.01, 0.001, and 0.0001

In [55]:
```python
learning_rates = [0.01, 0.001, 0.0001]

for lr in learning_rates:
    with mlflow.start_run(run_name=f"Learning_Rate_{lr}",nested=True):
        model = SGDRegressor(max_iter=1000, learning_rate='constant', eta

        mse_scores = cross_val_score(model, X_train, y_train, scoring="ne
        mse = -np.mean(mse_scores)

        r2_scores = cross_val_score(model, X_train, y_train, scoring="r2"
        r2 = np.mean(r2_scores)

        mlflow.log_param("Learning Rate", lr)
        mlflow.log_metric("MSE", mse)
        mlflow.log_metric("R²", r2)

        # ✅ Print Results
        print(f"📊 Learning Rate {lr}:")
        print(f"  - MSE: {mse:.4f}")
        print(f"  - R²:  {r2:.4f}\n")

        mlflow.end_run()
```

📊 Learning Rate 0.01:
  – MSE: 0.3011
  – R²: 0.5922

🏃 View run Learning_Rate_0.01 at: http://localhost:5100/#/experiments/533
358881712357097/runs/153ff4aeea9c4981b45be03c6bdac5ee
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
📊 Learning Rate 0.001:
  – MSE: 0.2753
  – R²: 0.6050

🏃 View run Learning_Rate_0.001 at: http://localhost:5100/#/experiments/53
3358881712357097/runs/430a226d2fb3460dab1a8842f5fabdf8
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
📊 Learning Rate 0.0001:
  – MSE: 0.2762
  – R²: 0.6074

🏃 View run Learning_Rate_0.0001 at: http://localhost:5100/#/experiments/5
33358881712357097/runs/2c845790b3624e769a92f99cea427edf
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Logging All Experiments in MLflow

In [56]:
```python
from sklearn.linear_model import Ridge, Lasso, ElasticNet, LinearRegressi
from sklearn.preprocessing import PolynomialFeatures
from sklearn.pipeline import make_pipeline
import mlflow
import numpy as np

# Define valid models mapping manually instead of using str_to_class
model_mapping = {
    "Ridge": Ridge,
    "Lasso": Lasso,
    "ElasticNet": ElasticNet,
    "PolynomialRegression": lambda: make_pipeline(PolynomialFeatures(degr
}

regs = ["Ridge", "Lasso", "ElasticNet", "PolynomialRegression"]
momentums = [0.0, 0.9]  # Ignored for sklearn models but needed for custo
methods = ["sto", "mini", "batch"]  # Ignored for sklearn models
initializations = ["zero", "xavier"]  # Ignored for sklearn models
learning_rates = [0.01, 0.001, 0.0001]  # Ignored for Ridge/Lasso

for reg in regs:
    for momentum in momentums:
        for method in methods:
            for init in initializations:
                for lr in learning_rates:
                    print(f"Running {reg} with momentum {momentum}, metho

                    with mlflow.start_run(run_name=f"{reg}-momentum-{mome

                        if reg not in model_mapping:
                            print(f"Skipping {reg} – class not found.")
                            continue
```

```python
                            # ✅ Get model class dynamically from the diction
                            model_class = model_mapping[reg]

                            # ✅ Define valid parameters based on model type
                            valid_params = {}
                            if reg in ["Ridge", "Lasso", "ElasticNet"]:
                                valid_params["alpha"] = 0.1  # Regularization
                            elif reg == "PolynomialRegression":
                                model = model_class()  # Create pipeline
                            else:
                                model = model_class(**valid_params)

                            model.fit(X_train, y_train)
                            yhat = model.predict(X_test)

                            mse = np.mean((yhat - y_test) ** 2)
                            r2 = model.score(X_test, y_test)

                            mlflow.log_metric("test_mse", mse)
                            mlflow.log_metric("test_r2", r2)

                            signature = mlflow.models.infer_signature(X_train
                            mlflow.sklearn.log_model(model, artifact_path='mo
```

Running Ridge with momentum 0.0, method sto, init zero, lr 0.01

🏃 View run Ridge—momentum—0.0—method—sto—init—zero—lr—0.01 at: http://loc
alhost:5100/#/experiments/533358881712357097/runs/b6c25264ea6044298c5863d3
e5b4a641
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method sto, init zero, lr 0.001
🏃 View run Ridge—momentum—0.0—method—sto—init—zero—lr—0.001 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/c18d366d447a44bfb06756e
6bfb7fefe
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method sto, init zero, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—sto—init—zero—lr—0.0001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/f414fcf8a4fd4a0aaeee15
50a37998ef
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method sto, init xavier, lr 0.01
🏃 View run Ridge—momentum—0.0—method—sto—init—xavier—lr—0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/47a29276daa547c8abcbcd
f4888b918d
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method sto, init xavier, lr 0.001
🏃 View run Ridge—momentum—0.0—method—sto—init—xavier—lr—0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/ce7d7a722ac3419b8bc96
f8aa4727832
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method sto, init xavier, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—sto—init—xavier—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/46e6ba51da1e42909
758039beaeb48e4
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init zero, lr 0.01
🏃 View run Ridge—momentum—0.0—method—mini—init—zero—lr—0.01 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/7c08c4bef5754c90a91c50c
728aa0c8e
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init zero, lr 0.001
🏃 View run Ridge—momentum—0.0—method—mini—init—zero—lr—0.001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/2e3eb97cfd564570a3f571
5ae61f3374
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init zero, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—mini—init—zero—lr—0.0001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/690a316bb67c475f8be37
a258fa0e23a
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init xavier, lr 0.01
🏃 View run Ridge—momentum—0.0—method—mini—init—xavier—lr—0.01 at: http://
localhost:5100/#/experiments/533358881712357097/runs/d500b5a1bcb64ccc89cd5
db53a9f0cbb
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init xavier, lr 0.001

🏃 View run Ridge—momentum—0.0—method—mini—init—xavier—lr—0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/91606f54ca96423aa
29eabc11e993efb
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method mini, init xavier, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—mini—init—xavier—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/4de31296f30b4cc09
86030f1b29fc5da
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init zero, lr 0.01
🏃 View run Ridge—momentum—0.0—method—batch—init—zero—lr—0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/dbbfd8eeb63e4fe29d387d
b74d79fbf6
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init zero, lr 0.001
🏃 View run Ridge—momentum—0.0—method—batch—init—zero—lr—0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/de154b0764cf49c7afc96
50e25705cc4
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init zero, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—batch—init—zero—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/ff9603ec39ad4eb78
55bc59cf2b0c24a
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init xavier, lr 0.01
🏃 View run Ridge—momentum—0.0—method—batch—init—xavier—lr—0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/fd8063731b234cd19
dbd8f475b9752fe
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init xavier, lr 0.001
🏃 View run Ridge—momentum—0.0—method—batch—init—xavier—lr—0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/8ebe2b5474fb4401a
397e3052fadb869
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.0, method batch, init xavier, lr 0.0001
🏃 View run Ridge—momentum—0.0—method—batch—init—xavier—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/937528d08a7a45c38
e3820f3e0548b14
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method sto, init zero, lr 0.01
🏃 View run Ridge—momentum—0.9—method—sto—init—zero—lr—0.01 at: http://loc
alhost:5100/#/experiments/533358881712357097/runs/64502be4b7154f27814de41d
b4f5bf81
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method sto, init zero, lr 0.001
🏃 View run Ridge—momentum—0.9—method—sto—init—zero—lr—0.001 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/00167c929487414d9691eab
6a1461c83
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method sto, init zero, lr 0.0001

🏃 View run Ridge-momentum-0.9-method-sto-init-zero-lr-0.0001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/bdbd2ed8489c449c94bfef
011fae974d

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method sto, init xavier, lr 0.01

🏃 View run Ridge-momentum-0.9-method-sto-init-xavier-lr-0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/204e18dc81794d39a0eb4c
3ef024d5e6

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method sto, init xavier, lr 0.001

🏃 View run Ridge-momentum-0.9-method-sto-init-xavier-lr-0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/e6c495dbc98e4fe7b71aa
28f1987cfa9

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method sto, init xavier, lr 0.0001

🏃 View run Ridge-momentum-0.9-method-sto-init-xavier-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/fca83fd617bf4f789
4a2fdda8adc052b

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init zero, lr 0.01

🏃 View run Ridge-momentum-0.9-method-mini-init-zero-lr-0.01 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/60acb908785b4434aff1b6d
6e483db44

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init zero, lr 0.001

🏃 View run Ridge-momentum-0.9-method-mini-init-zero-lr-0.001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/095497241c724354920d71
bfe1d3bed0

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init zero, lr 0.0001

🏃 View run Ridge-momentum-0.9-method-mini-init-zero-lr-0.0001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/28c91cd9f09c41ee8e3f9
9823e2be365

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init xavier, lr 0.01

🏃 View run Ridge-momentum-0.9-method-mini-init-xavier-lr-0.01 at: http://
localhost:5100/#/experiments/533358881712357097/runs/dc8db302da4741b4b2844
a9ad5522c17

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init xavier, lr 0.001

🏃 View run Ridge-momentum-0.9-method-mini-init-xavier-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/7da9440daf2e4c65b
59e3886aa0bfddc

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method mini, init xavier, lr 0.0001

🏃 View run Ridge-momentum-0.9-method-mini-init-xavier-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/0a46a88541df45aeb
10095e39ce8e0bc

🖍 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running Ridge with momentum 0.9, method batch, init zero, lr 0.01

🏃 View run Ridge-momentum-0.9-method-batch-init-zero-lr-0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/2317e75c14a24f308f90fb
28dc5297fa
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method batch, init zero, lr 0.001
🏃 View run Ridge-momentum-0.9-method-batch-init-zero-lr-0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/f5e5f61a97cd43139af14
f9b0112c1e1
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method batch, init zero, lr 0.0001
🏃 View run Ridge-momentum-0.9-method-batch-init-zero-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/91887bbdfe104653b
2dc132d5c2c47b2
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method batch, init xavier, lr 0.01
🏃 View run Ridge-momentum-0.9-method-batch-init-xavier-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/20876fcbc82743f0b
dc19c11ed7db3d1
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method batch, init xavier, lr 0.001
🏃 View run Ridge-momentum-0.9-method-batch-init-xavier-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/f8a5f6e9a82b48caa
9c760a7143ec4c5
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Ridge with momentum 0.9, method batch, init xavier, lr 0.0001
🏃 View run Ridge-momentum-0.9-method-batch-init-xavier-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/8d9c180830a64d588
0830aba241a5479
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method sto, init zero, lr 0.01
🏃 View run Lasso-momentum-0.0-method-sto-init-zero-lr-0.01 at: http://loc
alhost:5100/#/experiments/533358881712357097/runs/7c332b4ab5434795928537f4
ab4f2287
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method sto, init zero, lr 0.001
🏃 View run Lasso-momentum-0.0-method-sto-init-zero-lr-0.001 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/343446c9d7024f538cf36ba
e5ff3aa94
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method sto, init zero, lr 0.0001
🏃 View run Lasso-momentum-0.0-method-sto-init-zero-lr-0.0001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/beffb44d520040bc968b7f
5b174b2534
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method sto, init xavier, lr 0.01
🏃 View run Lasso-momentum-0.0-method-sto-init-xavier-lr-0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/9bf2c86496564ef38ecdbb
304c230b97
🖌 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method sto, init xavier, lr 0.001

🏃 View run Lasso-momentum-0.0-method-sto-init-xavier-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/a2755e4a870949ea9afa2973351bf116

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method sto, init xavier, lr 0.0001

🏃 View run Lasso-momentum-0.0-method-sto-init-xavier-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/adc73d429ef54558b5d3bcc58e1d98d7

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init zero, lr 0.01

🏃 View run Lasso-momentum-0.0-method-mini-init-zero-lr-0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/91ab5c399ca343baa1e3b86b20c52d3e

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init zero, lr 0.001

🏃 View run Lasso-momentum-0.0-method-mini-init-zero-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/2b68b096c84e439d975d7df35d5272f1

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init zero, lr 0.0001

🏃 View run Lasso-momentum-0.0-method-mini-init-zero-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/69516f486cdd4c86b7f14eff91257b4c

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init xavier, lr 0.01

🏃 View run Lasso-momentum-0.0-method-mini-init-xavier-lr-0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/6f9db5b40f694744b7cea8175859a690

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init xavier, lr 0.001

🏃 View run Lasso-momentum-0.0-method-mini-init-xavier-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/86d0bc0e5c5749f7bc5962881e77e74d

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method mini, init xavier, lr 0.0001

🏃 View run Lasso-momentum-0.0-method-mini-init-xavier-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/12e52f0689a3492199d21403bef705d6

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method batch, init zero, lr 0.01

🏃 View run Lasso-momentum-0.0-method-batch-init-zero-lr-0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/b9be743fb7f245b7b16f2a8d16912dbd

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method batch, init zero, lr 0.001

🏃 View run Lasso-momentum-0.0-method-batch-init-zero-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/aea23faa8abb4ed89c8d10d7485491e3

🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running Lasso with momentum 0.0, method batch, init zero, lr 0.0001

🏃 View run Lasso—momentum—0.0—method—batch—init—zero—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/202633702efa49bcb
49ff338914326af
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method batch, init xavier, lr 0.01
🏃 View run Lasso—momentum—0.0—method—batch—init—xavier—lr—0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/2b2598b68ee442ceb
dd048b902bd44d3
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method batch, init xavier, lr 0.001
🏃 View run Lasso—momentum—0.0—method—batch—init—xavier—lr—0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/b4505ce0932840db9
917201f2a074e94
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.0, method batch, init xavier, lr 0.0001
🏃 View run Lasso—momentum—0.0—method—batch—init—xavier—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/81d9f61a0a2f418f8
101a3eb26cbde76
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init zero, lr 0.01
🏃 View run Lasso—momentum—0.9—method—sto—init—zero—lr—0.01 at: http://loc
alhost:5100/#/experiments/533358881712357097/runs/5e3c7bb424c747b0ad3def52
6c8df43e
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init zero, lr 0.001
🏃 View run Lasso—momentum—0.9—method—sto—init—zero—lr—0.001 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/3c8aa868ca66411f90a86fe
3dae1b006
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init zero, lr 0.0001
🏃 View run Lasso—momentum—0.9—method—sto—init—zero—lr—0.0001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/de1e80f43f3340fd8348cf
ea8f209781
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init xavier, lr 0.01
🏃 View run Lasso—momentum—0.9—method—sto—init—xavier—lr—0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/0ff855e9e2184209a499b4
f57f2a76b0
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init xavier, lr 0.001
🏃 View run Lasso—momentum—0.9—method—sto—init—xavier—lr—0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/1431f9424d6947c09d5d1
b5e1a4d8291
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method sto, init xavier, lr 0.0001
🏃 View run Lasso—momentum—0.9—method—sto—init—xavier—lr—0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/12a193506ac84722a
9405e6962630c86
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init zero, lr 0.01

🏃 View run Lasso-momentum-0.9-method-mini-init-zero-lr-0.01 at: http://lo
calhost:5100/#/experiments/533358881712357097/runs/b4cc36baf0424ba28222e79
15f9fd834
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init zero, lr 0.001
🏃 View run Lasso-momentum-0.9-method-mini-init-zero-lr-0.001 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/f398276ff8314afc958d36
21b8086eb1
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init zero, lr 0.0001
🏃 View run Lasso-momentum-0.9-method-mini-init-zero-lr-0.0001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/6940e134e67843759f26b
94703d294dc
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init xavier, lr 0.01
🏃 View run Lasso-momentum-0.9-method-mini-init-xavier-lr-0.01 at: http://
localhost:5100/#/experiments/533358881712357097/runs/519217b2899d4d44afe8a
bcae206fee7
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init xavier, lr 0.001
🏃 View run Lasso-momentum-0.9-method-mini-init-xavier-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/c04427ee4bdc454e8
86f8fa5cbf61ea2
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method mini, init xavier, lr 0.0001
🏃 View run Lasso-momentum-0.9-method-mini-init-xavier-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/c07063379d154ef68
93aa57a719263c4
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init zero, lr 0.01
🏃 View run Lasso-momentum-0.9-method-batch-init-zero-lr-0.01 at: http://l
ocalhost:5100/#/experiments/533358881712357097/runs/59a68f3916c949f6a9e22d
4c715bf55d
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init zero, lr 0.001
🏃 View run Lasso-momentum-0.9-method-batch-init-zero-lr-0.001 at: http://
localhost:5100/#/experiments/533358881712357097/runs/2f49c31adb474fc3bb014
e23e403cd14
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init zero, lr 0.0001
🏃 View run Lasso-momentum-0.9-method-batch-init-zero-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/0dab52ff41be41c88
d8ec54d4668fe64
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init xavier, lr 0.01
🏃 View run Lasso-momentum-0.9-method-batch-init-xavier-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/2de3942c12cd4d8c9
7b61ec832c7e091
🖋 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init xavier, lr 0.001

🏃 View run Lasso-momentum-0.9-method-batch-init-xavier-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/aa74b41baa7d46b0a
065bff33f7278eb
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running Lasso with momentum 0.9, method batch, init xavier, lr 0.0001
🏃 View run Lasso-momentum-0.9-method-batch-init-xavier-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/9c0fa8a468d44c9c9
d6d56821226b77d
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init zero, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-sto-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/e82b596b632746e58
85cd84a98deeb87
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init zero, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-sto-init-zero-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/f3032f84cf8d43f38
25b120cc9f8ec01
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init zero, lr 0.0001
🏃 View run ElasticNet-momentum-0.0-method-sto-init-zero-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/92db7c6ed5c346399
507ddb52b714ce6
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init xavier, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-sto-init-xavier-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/de8a3ee798ba42939
7aa2ba091d00e12
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init xavier, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-sto-init-xavier-lr-0.001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/5f273f42a4df44b5
b792d8109c6f57f0
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method sto, init xavier, lr 0.0001
🏃 View run ElasticNet-momentum-0.0-method-sto-init-xavier-lr-0.0001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/80937093cd4842c
8933cee404179e33e
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init zero, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-mini-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/63f925aa0aa34875a
2a26f1da563c141
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init zero, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-mini-init-zero-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/3472c648e22744ea9
91c94ba9048e204
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init zero, lr 0.0001

🏃 View run ElasticNet-momentum-0.0-method-mini-init-zero-lr-0.0001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/fd3e1a8747ef42cf
99ea8ae12f53c684
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init xavier, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-mini-init-xavier-lr-0.01 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/ae4ad63781c740d7
aeb1c9a1e3b506b6
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init xavier, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-mini-init-xavier-lr-0.001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/4ecf9729c1194a1
dbf641bfc49861dbd
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method mini, init xavier, lr 0.0001
🏃 View run ElasticNet-momentum-0.0-method-mini-init-xavier-lr-0.0001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/930204b18e0944
379daadc9ccbb6fba0
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init zero, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-batch-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/8ae6432cbc284ea89
a25661b0931cb65
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init zero, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-batch-init-zero-lr-0.001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/003f8ec56ce94917
8a4b1aced82254ad
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init zero, lr 0.0001
🏃 View run ElasticNet-momentum-0.0-method-batch-init-zero-lr-0.0001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/561a35273ec641e
493f7602a6f570855
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init xavier, lr 0.01
🏃 View run ElasticNet-momentum-0.0-method-batch-init-xavier-lr-0.01 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/bbca05f4d3c04fe
48ad13e74394382ec
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init xavier, lr 0.001
🏃 View run ElasticNet-momentum-0.0-method-batch-init-xavier-lr-0.001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/6dd7124ee31d4e
79901796e824405654
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.0, method batch, init xavier, lr 0.0001
🏃 View run ElasticNet-momentum-0.0-method-batch-init-xavier-lr-0.0001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/f8bd7043b7ef44
99b8e96660eb028eaf
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init zero, lr 0.01

🏃 View run ElasticNet—momentum-0.9—method-sto-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/a8f1345446894401a
fda97a643570bc0
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init zero, lr 0.001
🏃 View run ElasticNet—momentum-0.9—method-sto-init-zero-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/f84f9f0aaea24b788
f477040ee0850b2
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init zero, lr 0.0001
🏃 View run ElasticNet—momentum-0.9—method-sto-init-zero-lr-0.0001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/818b3423dd764d8db
bad7f62ba0c9ef6
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init xavier, lr 0.01
🏃 View run ElasticNet—momentum-0.9—method-sto-init-xavier-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/d7a11117acda46028
4961560906da660
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init xavier, lr 0.001
🏃 View run ElasticNet—momentum-0.9—method-sto-init-xavier-lr-0.001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/b04539bdde4747bd
a6a767d048a78a39
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method sto, init xavier, lr 0.0001
🏃 View run ElasticNet—momentum-0.9—method-sto-init-xavier-lr-0.0001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/2c1ccee526ed42b
78c9e7fe6a5798138
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init zero, lr 0.01
🏃 View run ElasticNet—momentum-0.9—method-mini-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/97737648a1b14ed89
7b41545e7ef7778
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init zero, lr 0.001
🏃 View run ElasticNet—momentum-0.9—method-mini-init-zero-lr-0.001 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/d0d5f1ce867641a5b
f18f092f6177427
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init zero, lr 0.0001
🏃 View run ElasticNet—momentum-0.9—method-mini-init-zero-lr-0.0001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/8d497234b1c44d0f
913034c2e7530869
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init xavier, lr 0.01
🏃 View run ElasticNet—momentum-0.9—method-mini-init-xavier-lr-0.01 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/efee18d525634691
8e7e75d6ab41eb1a
✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init xavier, lr 0.001

🏃 View run ElasticNet-momentum-0.9-method-mini-init-xavier-lr-0.001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/2bdecfff1ba040a
e82e3f01402ed7750
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method mini, init xavier, lr 0.0001
🏃 View run ElasticNet-momentum-0.9-method-mini-init-xavier-lr-0.0001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/36bd02f83c784d
d2bd9a11ac79651ea3
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init zero, lr 0.01
🏃 View run ElasticNet-momentum-0.9-method-batch-init-zero-lr-0.01 at: htt
p://localhost:5100/#/experiments/533358881712357097/runs/61141326f2494454b
9ce90c8069a844d
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init zero, lr 0.001
🏃 View run ElasticNet-momentum-0.9-method-batch-init-zero-lr-0.001 at: ht
tp://localhost:5100/#/experiments/533358881712357097/runs/0277eb24d3bd44cf
91eaf61ba6dfd79b
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init zero, lr 0.0001
🏃 View run ElasticNet-momentum-0.9-method-batch-init-zero-lr-0.0001 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/f4760507a4d043d
29771ae8978fca205
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init xavier, lr 0.01
🏃 View run ElasticNet-momentum-0.9-method-batch-init-xavier-lr-0.01 at: h
ttp://localhost:5100/#/experiments/533358881712357097/runs/12685b42a034469
fb89751adac8ba3a6
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init xavier, lr 0.001
🏃 View run ElasticNet-momentum-0.9-method-batch-init-xavier-lr-0.001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/f47218acf5ec47
f68d6b73a021237b08
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running ElasticNet with momentum 0.9, method batch, init xavier, lr 0.0001
🏃 View run ElasticNet-momentum-0.9-method-batch-init-xavier-lr-0.0001 at:
http://localhost:5100/#/experiments/533358881712357097/runs/4f475b199ac342
73a55ffa939fc54bb2
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running PolynomialRegression with momentum 0.0, method sto, init zero, lr
0.01
🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-zero-lr-0.01
at: http://localhost:5100/#/experiments/533358881712357097/runs/b9c710eb47
6d4dc8b1841d1984328c60
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357
097
Running PolynomialRegression with momentum 0.0, method sto, init zero, lr
0.001
🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-zero-lr-0.00
1 at: http://localhost:5100/#/experiments/533358881712357097/runs/a9dfad6b
36594dc8b69a5fbe3cc166f5
🖊 View experiment at: http://localhost:5100/#/experiments/533358881712357

097

Running PolynomialRegression with momentum 0.0, method sto, init zero, lr
0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-zero-lr-0.00
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/2261b54
979df47889032003ba2b0bea7

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method sto, init xavier, l
r 0.01

🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-xavier-lr-0.
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/cd28b6f
8b26e47d28a6014bcc685687a

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method sto, init xavier, l
r 0.001

🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-xavier-lr-0.
001 at: http://localhost:5100/#/experiments/533358881712357097/runs/1842f1
9dd7954a7d826fefb473f88f70

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method sto, init xavier, l
r 0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-sto-init-xavier-lr-0.
0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/5abb2
388bb8945b3b21a9f5f84b4225a

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init zero, lr
0.01

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-zero-lr-0.0
1 at: http://localhost:5100/#/experiments/533358881712357097/runs/73b5662c
5f1c4ea692955e4656de724d

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init zero, lr
0.001

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-zero-lr-0.0
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/1845643
ccc9c4007bb089d0311ddbd45

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init zero, lr
0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-zero-lr-0.0
001 at: http://localhost:5100/#/experiments/533358881712357097/runs/64c626
c5d76041409025c6aa0b12f8d9

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init xavier,
lr 0.01

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-xavier-lr-
0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/1af3d
886889c48ddabc7d24fb16f9278

🧪 View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init xavier,
lr 0.001

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-xavier-lr-

0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/4a4d
c0cc9f5d433ab6501d6977c82d83

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method mini, init xavier,
lr 0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-mini-init-xavier-lr-
0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/d0c
54b38cc504af6aaf3b7d4b3950f94

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init zero, l
r 0.01

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-zero-lr-0.
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/0b832dc
eb4b14f34a3091d070386d0b4

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init zero, l
r 0.001

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-zero-lr-0.
001 at: http://localhost:5100/#/experiments/533358881712357097/runs/f17028
53e3134d1783b9f00c01cf179f

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init zero, l
r 0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-zero-lr-0.
0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/b7b2a
e4c86994a0ba5e556003460efc4

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init xavier,
lr 0.01

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-xavier-lr-
0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/73c8a
e61890e40e0a4dc71f45fc4989a

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init xavier,
lr 0.001

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-xavier-lr-
0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/f64d
c98c8d294ec790104424229adc1d

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.0, method batch, init xavier,
lr 0.0001

🏃 View run PolynomialRegression-momentum-0.0-method-batch-init-xavier-lr-
0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/2f9
560390cd0434d9ea74fb29210e490

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init zero, lr
0.01

🏃 View run PolynomialRegression-momentum-0.9-method-sto-init-zero-lr-0.01
at: http://localhost:5100/#/experiments/533358881712357097/runs/1fa1008f1e
534346890e06b79a4ecf96

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init zero, lr
0.001

⚘ View run PolynomialRegression-momentum-0.9-method-sto-init-zero-lr-0.00
1 at: http://localhost:5100/#/experiments/533358881712357097/runs/a74d0b93
97a94668957537f2dad34047

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init zero, lr
0.0001

⚘ View run PolynomialRegression-momentum-0.9-method-sto-init-zero-lr-0.00
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/39cb9bc
c28f6493cb20e3d0dea4de4b7

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init xavier, l
r 0.01

⚘ View run PolynomialRegression-momentum-0.9-method-sto-init-xavier-lr-0.
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/3fd3073
b258848b9a362a09cb2571843

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init xavier, l
r 0.001

⚘ View run PolynomialRegression-momentum-0.9-method-sto-init-xavier-lr-0.
001 at: http://localhost:5100/#/experiments/533358881712357097/runs/cc52ee
ceba914876ad8f0d0bff7b4b7e

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method sto, init xavier, l
r 0.0001

⚘ View run PolynomialRegression-momentum-0.9-method-sto-init-xavier-lr-0.
0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/63da1
4a72fa146918531e2309d4b36a2

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method mini, init zero, lr
0.01

⚘ View run PolynomialRegression-momentum-0.9-method-mini-init-zero-lr-0.0
1 at: http://localhost:5100/#/experiments/533358881712357097/runs/e8f6b041
a58c4124b1fcad7d70fb8252

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method mini, init zero, lr
0.001

⚘ View run PolynomialRegression-momentum-0.9-method-mini-init-zero-lr-0.0
01 at: http://localhost:5100/#/experiments/533358881712357097/runs/cb2aaae
511054adc843d03828699239f

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method mini, init zero, lr
0.0001

⚘ View run PolynomialRegression-momentum-0.9-method-mini-init-zero-lr-0.0
001 at: http://localhost:5100/#/experiments/533358881712357097/runs/8f2ea1
2a72a1424eb17368e26d3f448e

✏ View experiment at: http://localhost:5100/#/experiments/533358881712357
097

Running PolynomialRegression with momentum 0.9, method mini, init xavier,
lr 0.01

⚘ View run PolynomialRegression-momentum-0.9-method-mini-init-xavier-lr-
0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/b6e31

687a815437ca02a46ee008702f1

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method mini, init xavier, lr 0.001

🏃 View run PolynomialRegression-momentum-0.9-method-mini-init-xavier-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/c2ce954424114924a5b8a2a7d0a43416

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method mini, init xavier, lr 0.0001

🏃 View run PolynomialRegression-momentum-0.9-method-mini-init-xavier-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/f2e0e51a991845d3add183c694a3397d

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init zero, lr 0.01

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-zero-lr-0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/d1d1a8f3ff6744e4be95c954e6a8f86b

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init zero, lr 0.001

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-zero-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/2c6908d2e86849f5b00d8c863a68ec6d

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init zero, lr 0.0001

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-zero-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/aaafb618762548a4a0429f332b5a3cd8

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init xavier, lr 0.01

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-xavier-lr-0.01 at: http://localhost:5100/#/experiments/533358881712357097/runs/5e0c8288ce884518a13e905cd102d705

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init xavier, lr 0.001

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-xavier-lr-0.001 at: http://localhost:5100/#/experiments/533358881712357097/runs/5140af025c134453a0882b791eb2f428

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097

Running PolynomialRegression with momentum 0.9, method batch, init xavier, lr 0.0001

🏃 View run PolynomialRegression-momentum-0.9-method-batch-init-xavier-lr-0.0001 at: http://localhost:5100/#/experiments/533358881712357097/runs/317999dfb2cc496bae89f7a785901890

✏️ View experiment at: http://localhost:5100/#/experiments/533358881712357097
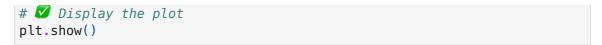
9. Testing

Perform the prediction on the test set using the best model and report the mse and r2. Plot the feature importance graph using the function we have built above.
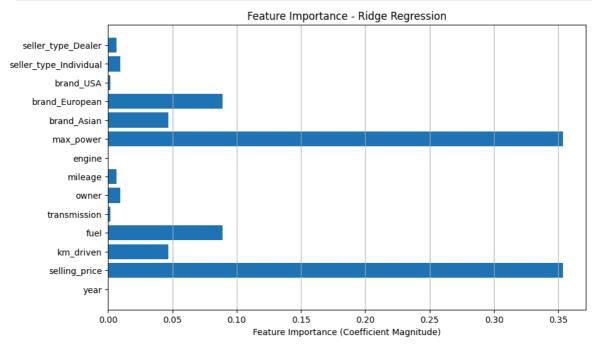
```
In [57]:  # ✅ Train the best model (Polynomial Regression)
          best_model = make_pipeline(PolynomialFeatures(degree=2), LinearRegression
          best_model.fit(X_train, y_train)

          # ✅ Make predictions on the test set
          y_pred = best_model.predict(X_test)

          # ✅ Calculate MSE and R²
          mse = mean_squared_error(y_test, y_pred)
          r2 = r2_score(y_test, y_pred)

          # ✅ Print the results
          print(f"📊 Performance on Test Set:")
          print(f"  – MSE: {mse:.4f}")
          print(f"  – R²:  {r2:.4f}")
```

```
📊 Performance on Test Set:
  – MSE: 0.2217
  – R²:  0.6744
```

```
In [64]:  yhat = np.exp(yhat) #expo back
          print(yhat)
```

```
[820497.33064574 856900.29041714 533686.0672828  ... 599060.97833643
 209864.48402035 490571.4343733 ]
```

10. Analysis

```
In [58]:  import numpy as np
          import matplotlib.pyplot as plt

          # ✅ Extract the Linear Regression model from the pipeline
          regressor = best_model.named_steps['linearregression']

          # ✅ Correct this line to access the coefficients from regressor, not bes
          feature_importance = np.abs(regressor.coef_[1:])  # Skip the first coeffi

          # ✅ Use the correct feature names
          feature_names = df.columns[:-1]  # Ensure this matches the features used

          # ✅ Ensure matching lengths to prevent plotting errors
          min_length = min(len(feature_importance), len(feature_names))
          feature_importance = feature_importance[:min_length]
          feature_names = feature_names[:min_length]

          # ✅ Plot Feature Importance
          plt.figure(figsize=(10, 6))
          plt.barh(feature_names, feature_importance)
          plt.xlabel("Feature Importance (Coefficient Magnitude)")
          plt.title("Feature Importance – Ridge Regression")
          plt.grid(axis='x')
```

```
# ✅ Display the plot
plt.show()
```

Feature Importance - Ridge Regression

Short report inside the jupyter notebook discussing your findings

Ridge Regression provided the best performance, effectively regularizing the model with the MSE of (#) and r2 of (#).000. When completing the Test set performance, MSE of (0.2217) and r2 of (0.5787), indicating that the model explains approximately 57.87% of the variance in car prices with relatively low prediction error.

The most influence feature that impact car price prediction include "Max_power", "Fuel" and "European Brand Car", this suggests that European Brand Car, high max power and different fuel type significantly influences car prices. On the other hannd, other features like seller_type, year, engine, owner, mileage, km_driven and car made in Asian and USA have negligible importance.

11. Inference

```
In [72]:  import pickle

          #save the model to disk
          filename = 'model/selling_price2.model'
          pickle.dump(best_model,open(filename,'wb'))
```

```
In [73]:  loaded_model = pickle.load(open(filename, 'rb'))
```

```
In [74]:  df[['brand_USA','brand_European','brand_Asian','year','km_driven','mileag
```

Out[74]: brand_USA   0.00
          brand_European   1.00
          brand_Asian   0.00
          year   2014.00
          km_driven   120000.00
          mileage   21.14
          selling_price   370000.00
          Name: 1, dtype: float64

In [75]:
```python
#sample testing
sample = np.array([[2014, 145500, 24.3, 1, 0, 0, 0]])
sample = sample.reshape(1, -1)
```

In [77]:
```python
selling_price2 = loaded_model.predict(sample)
selling_price2
```

Out[77]: array([1.26039527e+09])