

Our goal is to create a model

- predicting the probability that a borrower will prepay their mortgage
- fitting the model using the prior 10 years worth of data

Mortgage rates, with lag

To over-simplify

- we examine the probability of prepayment
- only for mortgages that are exactly 10 years old

We show the current mortgage rate (blue) and the borrower's actual rate (orange).

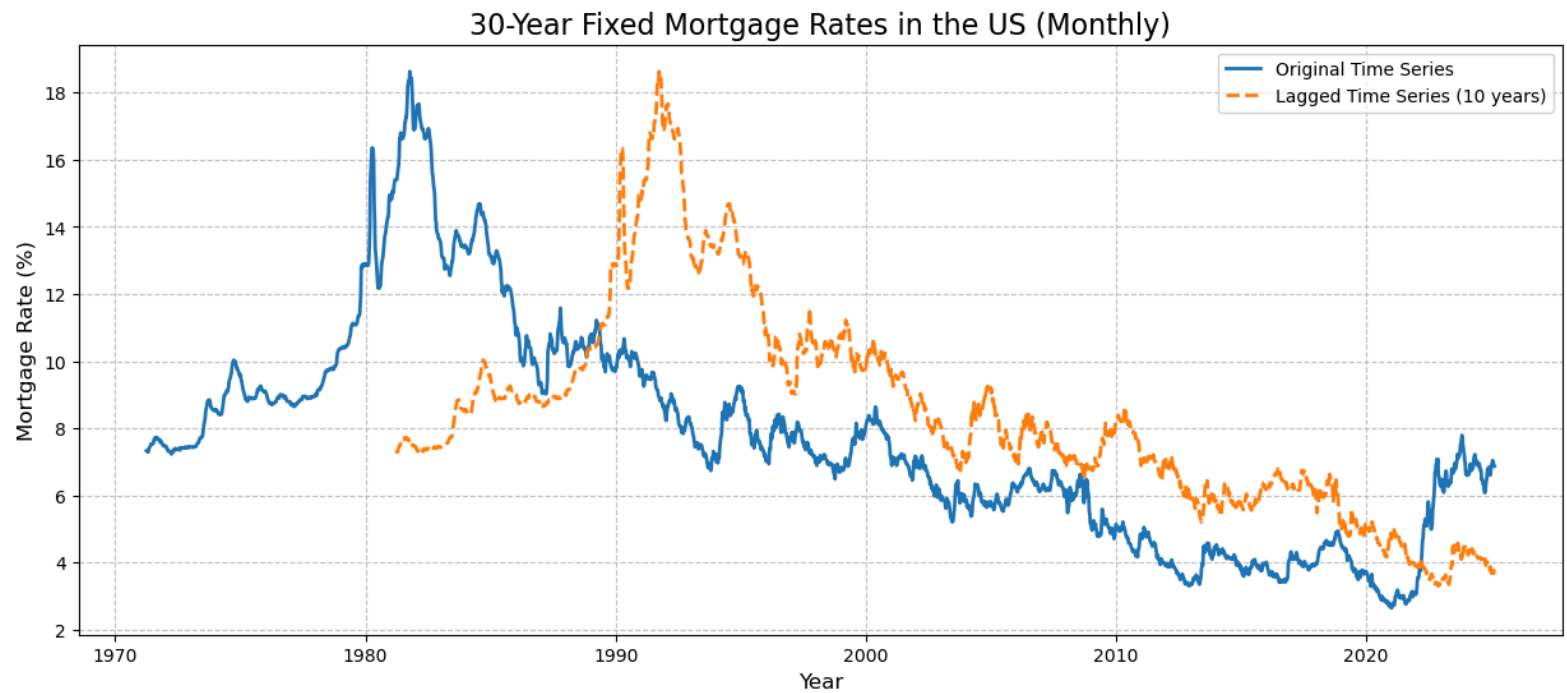
The borrower has an incentive to prepay

- when the current rate (blue)
- is **lower** than their actual rate (orange)

Here is the data.

```
In [4]: fig_levels
```

```
Out[4]:
```



Incentive: the semantics of prepayment

Let us create a synthetic feature

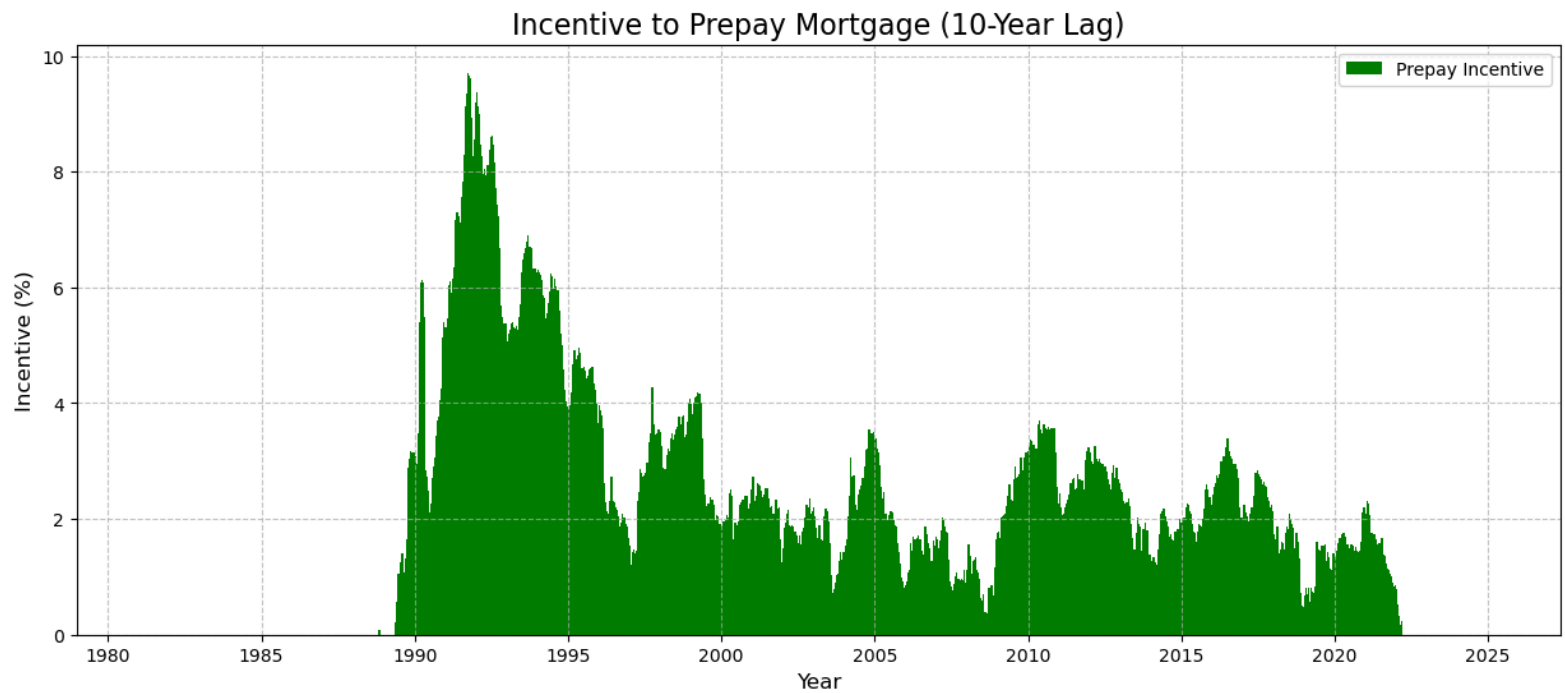
- the Incentive to prepay

This feature is useful both because

- it captures the **reason** (semantics) why a borrower might prepay
- and is **not** dependent on the level of rates

```
In [5]: fig_incentive
```

```
Out[5]:
```



**Incentive advantage: works even at times when
Current Rate is at extremes**

Imagine that we try to fit a model at each date

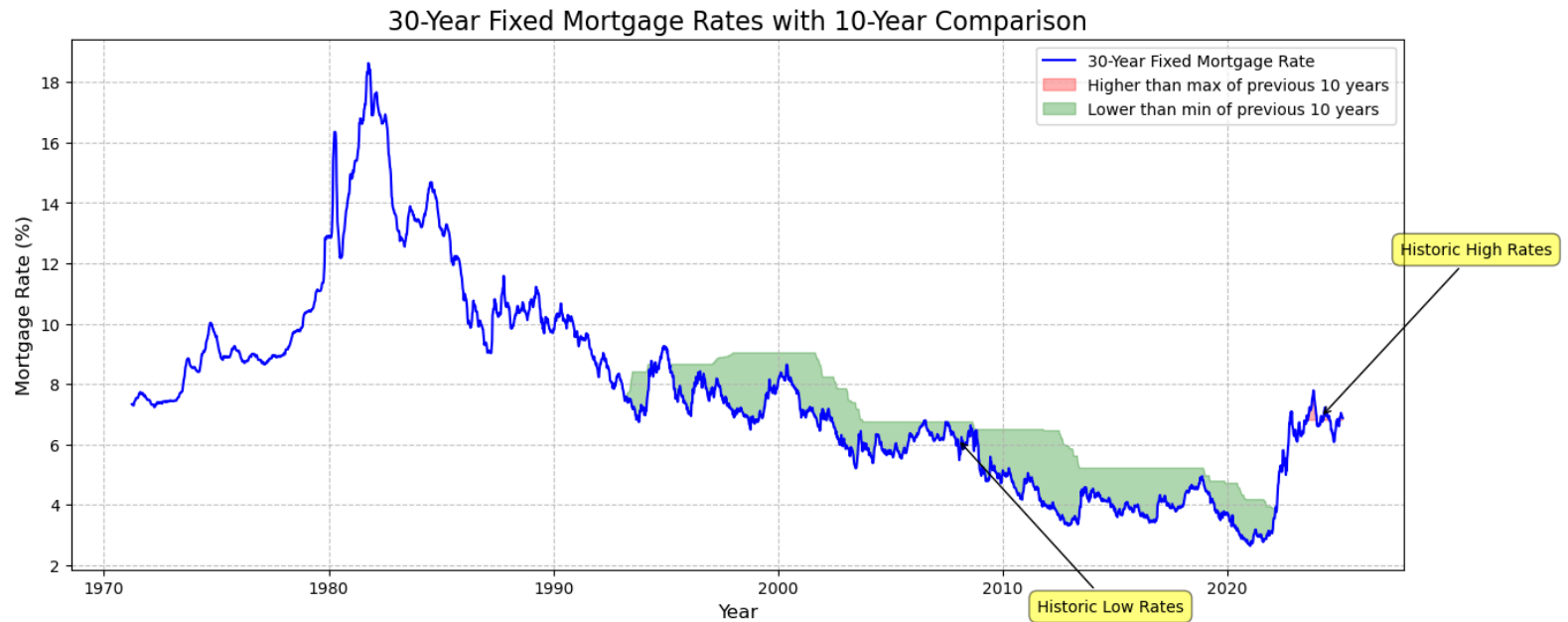
- using training data from the prior 10 years

There will be times (highlighted periods in following plot)

- when the current mortgage rate
- is either at a historic high (red shading) or low (green shading) relative to the training data

In [6]: fig_historic_low_high

Out[6]:



Using the raw features

- current mortgage rate
- borrower's actual rate

will violate the Fundamental Theorem of Machine Learning.

- the distribution of mortgage rates is not the same
 - for the training data (prior 10 years)
 - and the out of sample period

In [7]: `print("Done")`

Done

