Model Visualisation

Amit Kapoor
@amitkaps

Story

“We don’t see things as they are, we see them as we are.”

— *Anais Nin*

The Blind Men & the Elephant

___

The Blind Men & the Elephant

———

“And so these men of Indostan
Disputed loud and long,
Each in his own opinion
Exceeding stiff and strong,
Though each was partly in the right,
And all were in the wrong.”

— John Godfrey Saxe

The Elephant: Data

“Data is just a clue to the end truth”

— *Josh Smith*

The Men: Building Models

"All models are wrong, but some are useful"

— *George Box*

Layers of Abstraction

Data Abstraction

Visual Abstraction

Model Abstraction

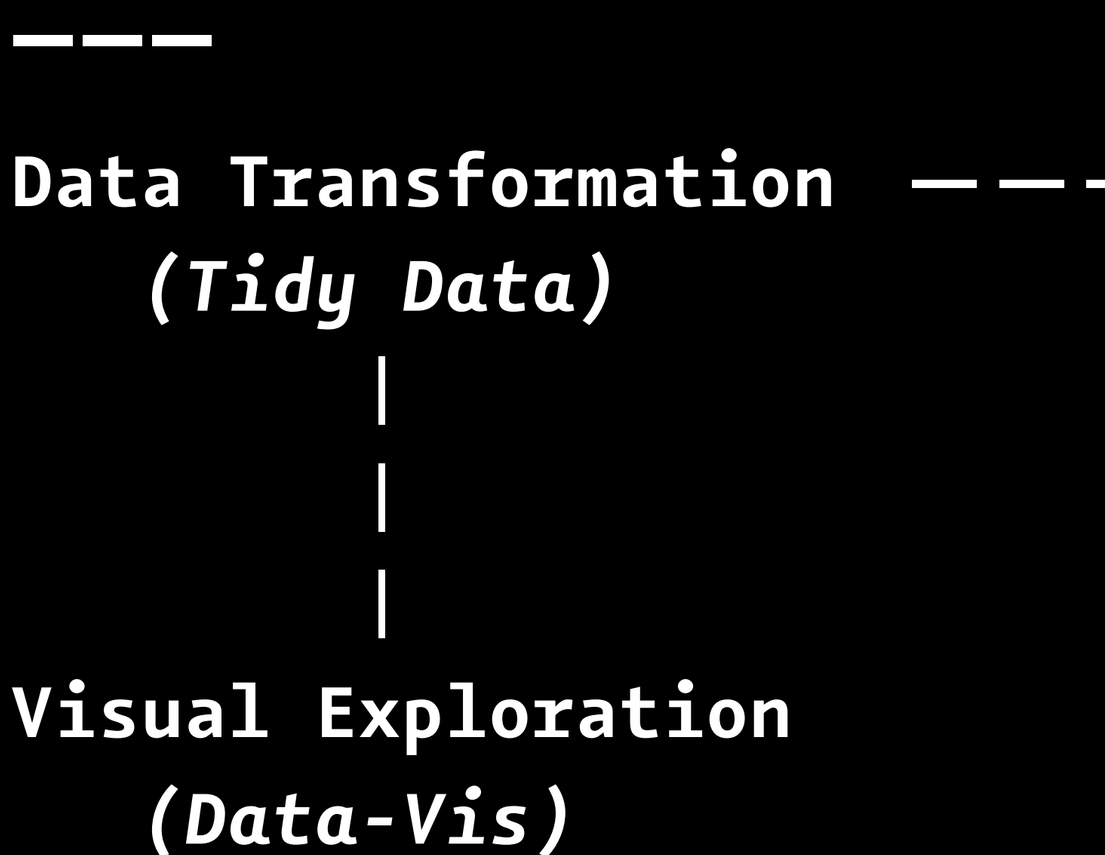
Machine Learning (ML) Speak

Data Transformation

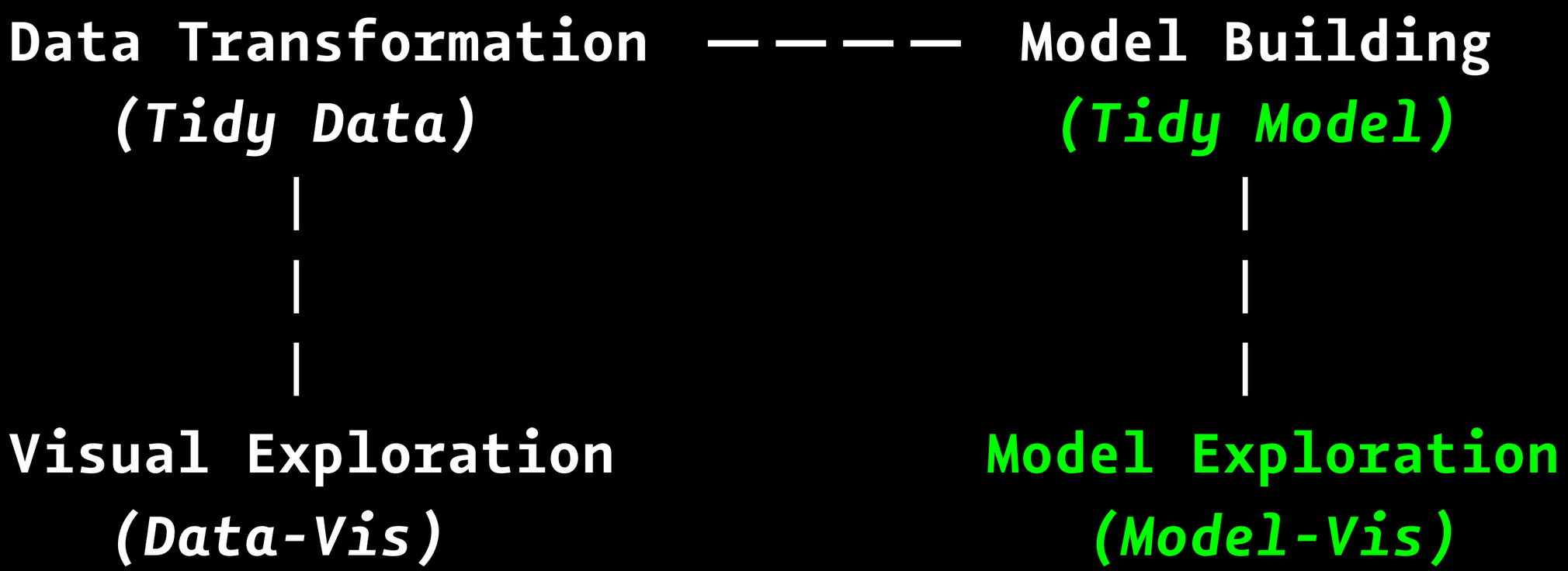
Visual Exploration

Model Building

ML Pipeline


Data Transformation
(Tidy Data)
Model Building
Visual Exploration
(Data-Vis)

ML Pipeline++


Data Transformation
(Tidy Data)
Model Building
(Tidy Model)
Visual Exploration
(Data-Vis)
Model Exploration
(Model-Vis)

Model Exploration

Use visualisation to aid the transition of **implicit knowledge** in the data and your head to **explicit knowledge** in the model.

Model Visualisation

Model-Vis approach

Tidy Model concept

Example: Buying a Car

___

Colloquially: “kitna deti hain”

Example: Buying a Car

Colloquially: “kitna deti hain”

Translates to “*What is the mileage, for the price I pay?*”

Cars Dataset

Scraped from comparison website

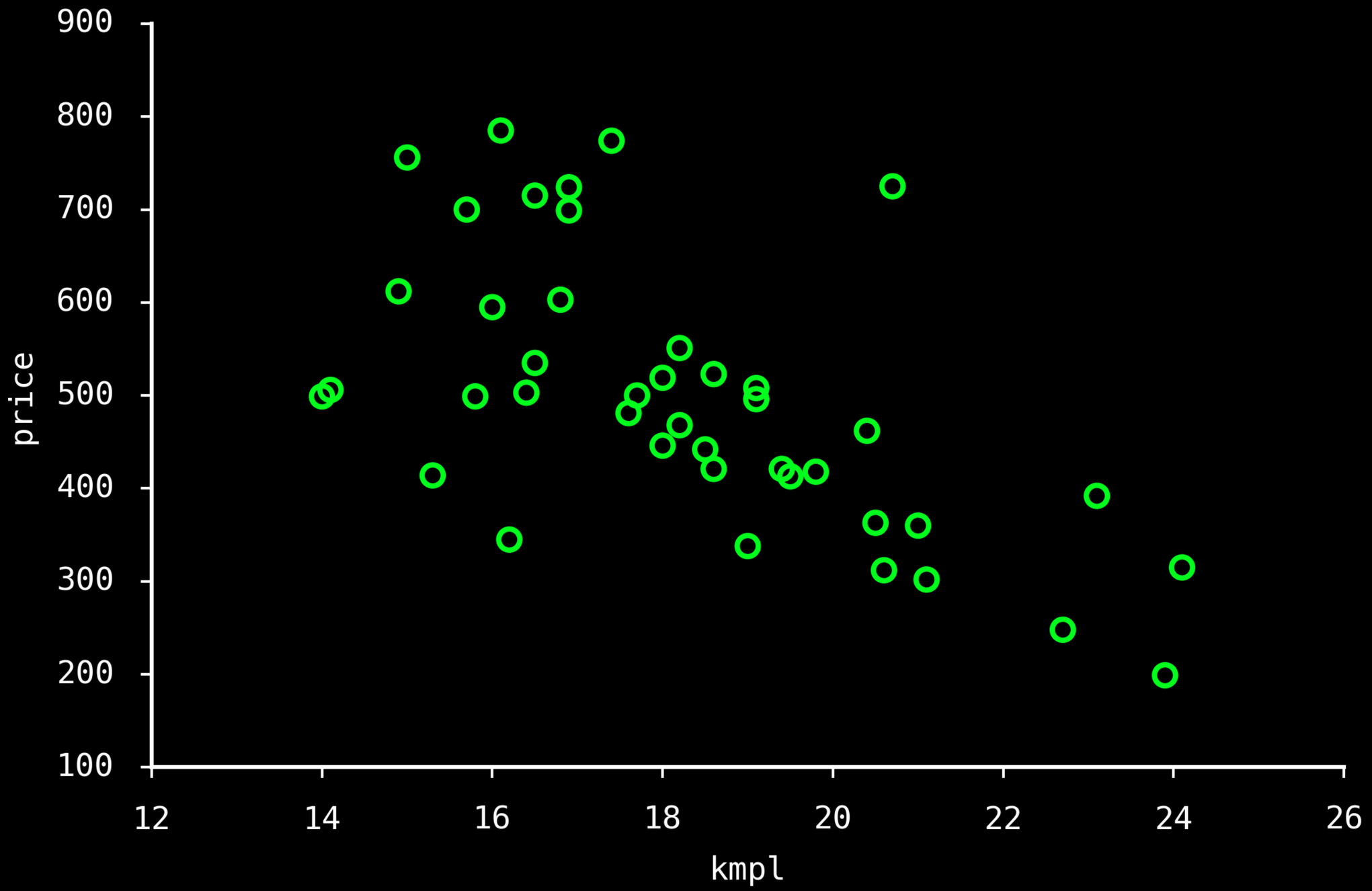
Refined & tidied up

Base version for petrol cars

Price < ₹ 1,000K

n = 42

brand	model	price	kmpl
Tata	Nano	199	23.9
Suzuki	Alto800	248	22.7
Hyundai	EON	302	21.1
Nissan	Datsun	312	20.6
...	...	...	...
Suzuki	Ciaz	725	20.7
Skoda	Rapid	756	15.0
Hyundai	Verna	774	17.4
VW	Vento	785	16.1


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Model-Vis Approach

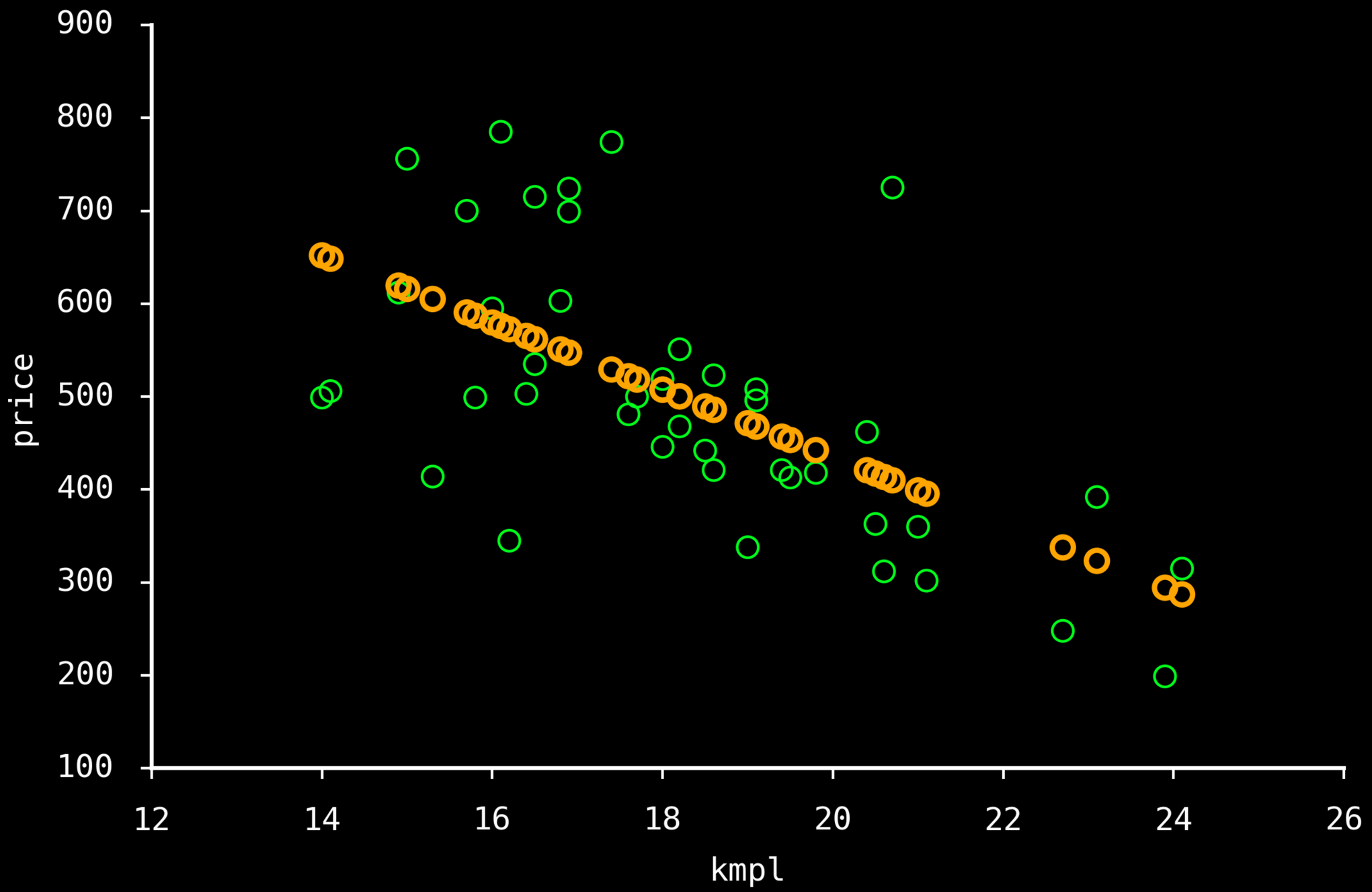
---

[o] Visualise the data space

Build a Simple Model

Regression: $price = \beta_0 + \beta_1 * kmpl$

Ordinary Least Square (OLS)


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Model-Vis Approach

---

[1] Visualise the predictions in the data space

Change Model Parameters?

Regression: $price = \beta_0 + \beta_1 * kmpl$

Ordinary Least Square

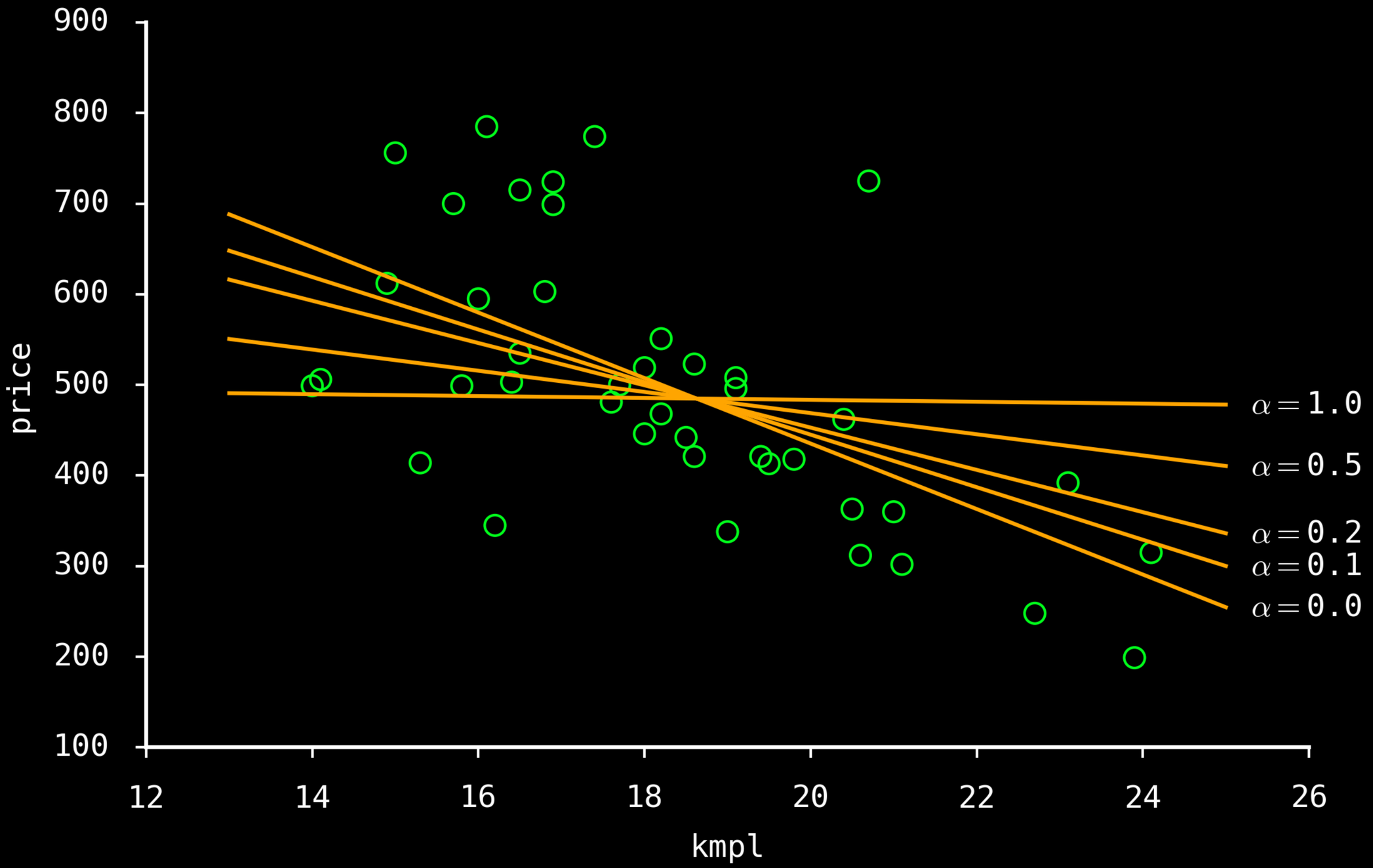
// One-Shot ML Algorithm //

Modify the Cost Function

Regression: $price = \beta_0 + \beta_1 * kmpl$

~~Ordinary Least Square~~

Ridge: L2 Regularisation


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl
$\alpha = 1.0$
$\alpha = 0.5$
$\alpha = 0.2$
$\alpha = 0.1$
$\alpha = 0.0$

Model-Vis Approach

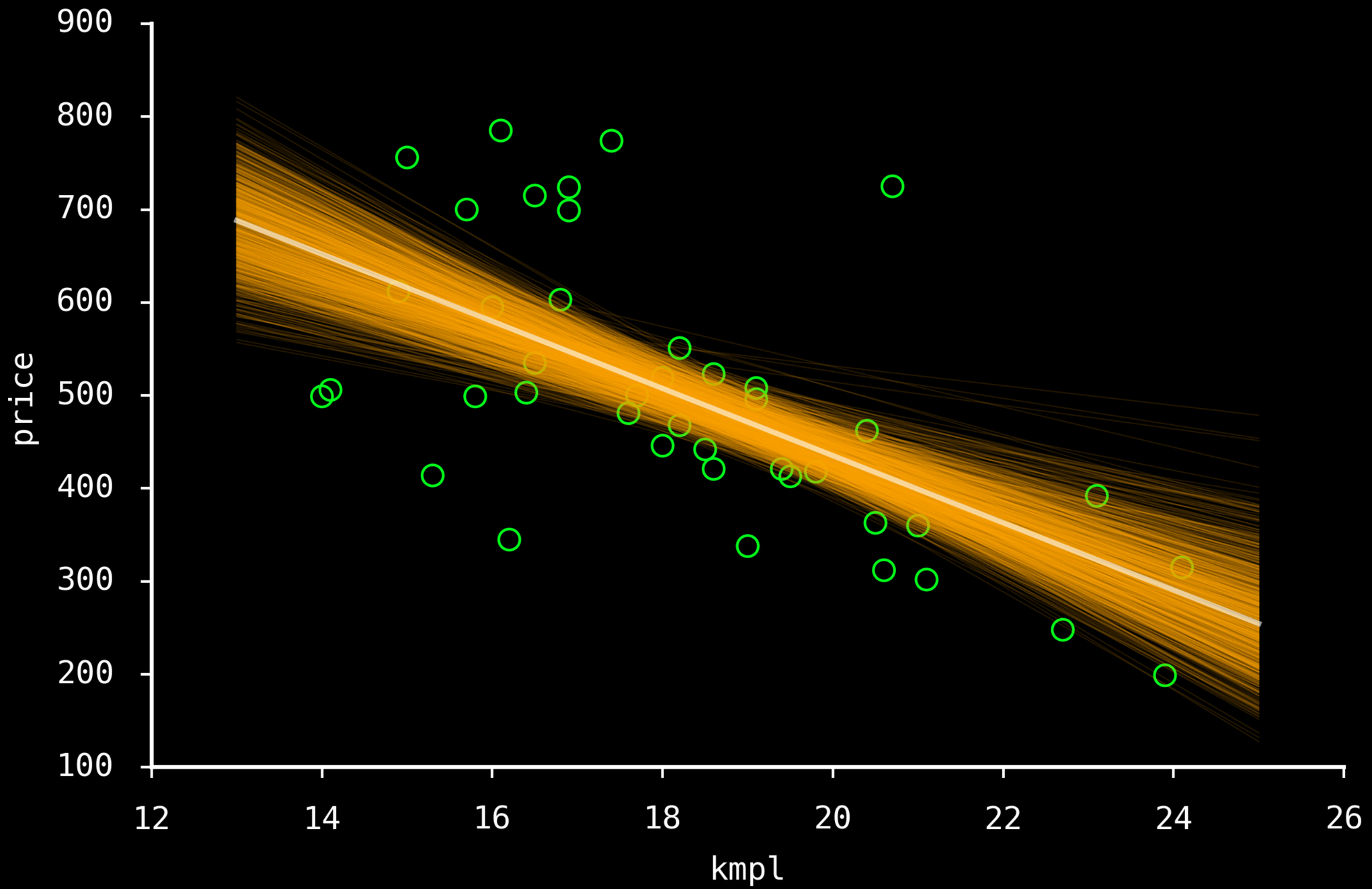
---

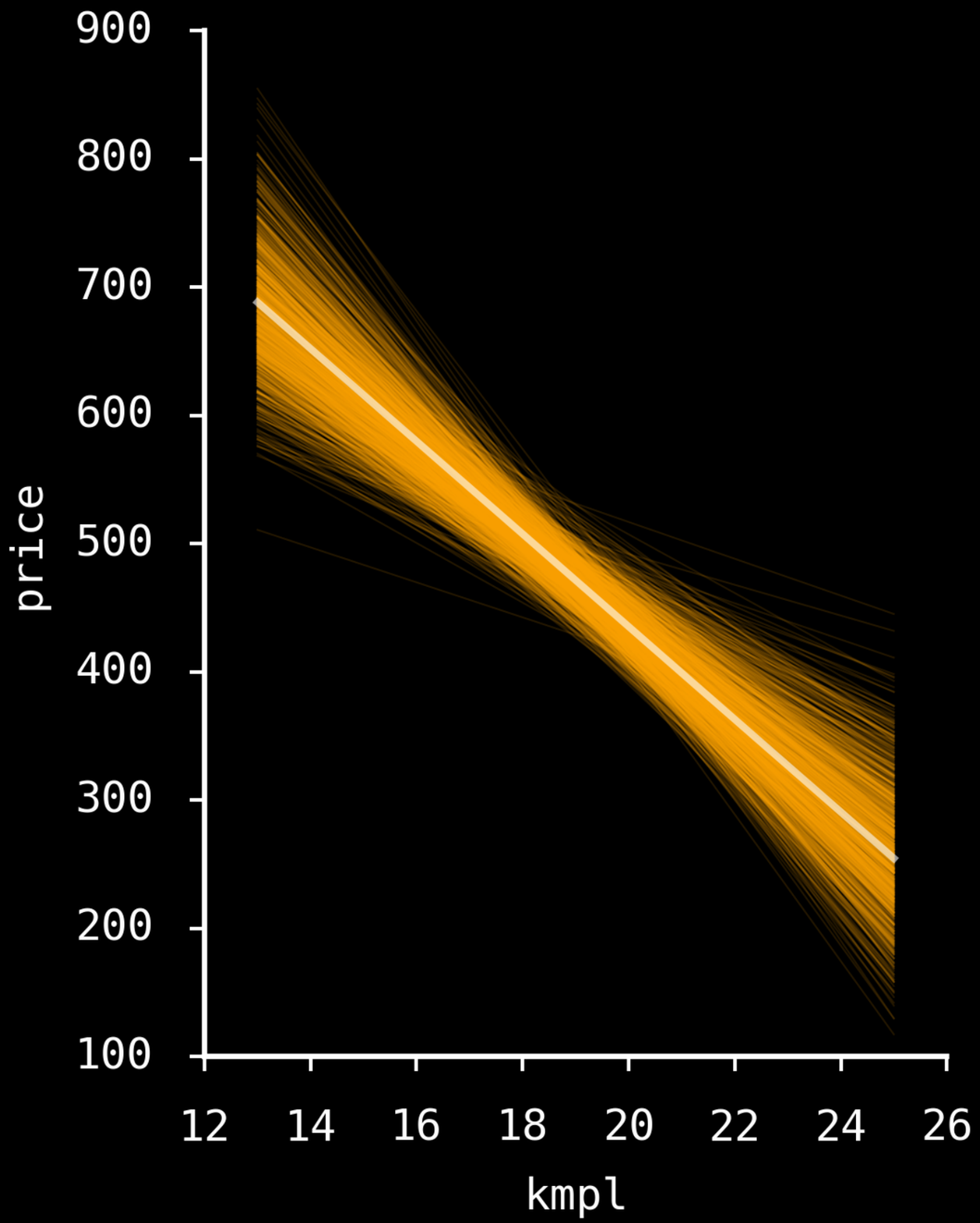
[2] Visualise with different model parameters

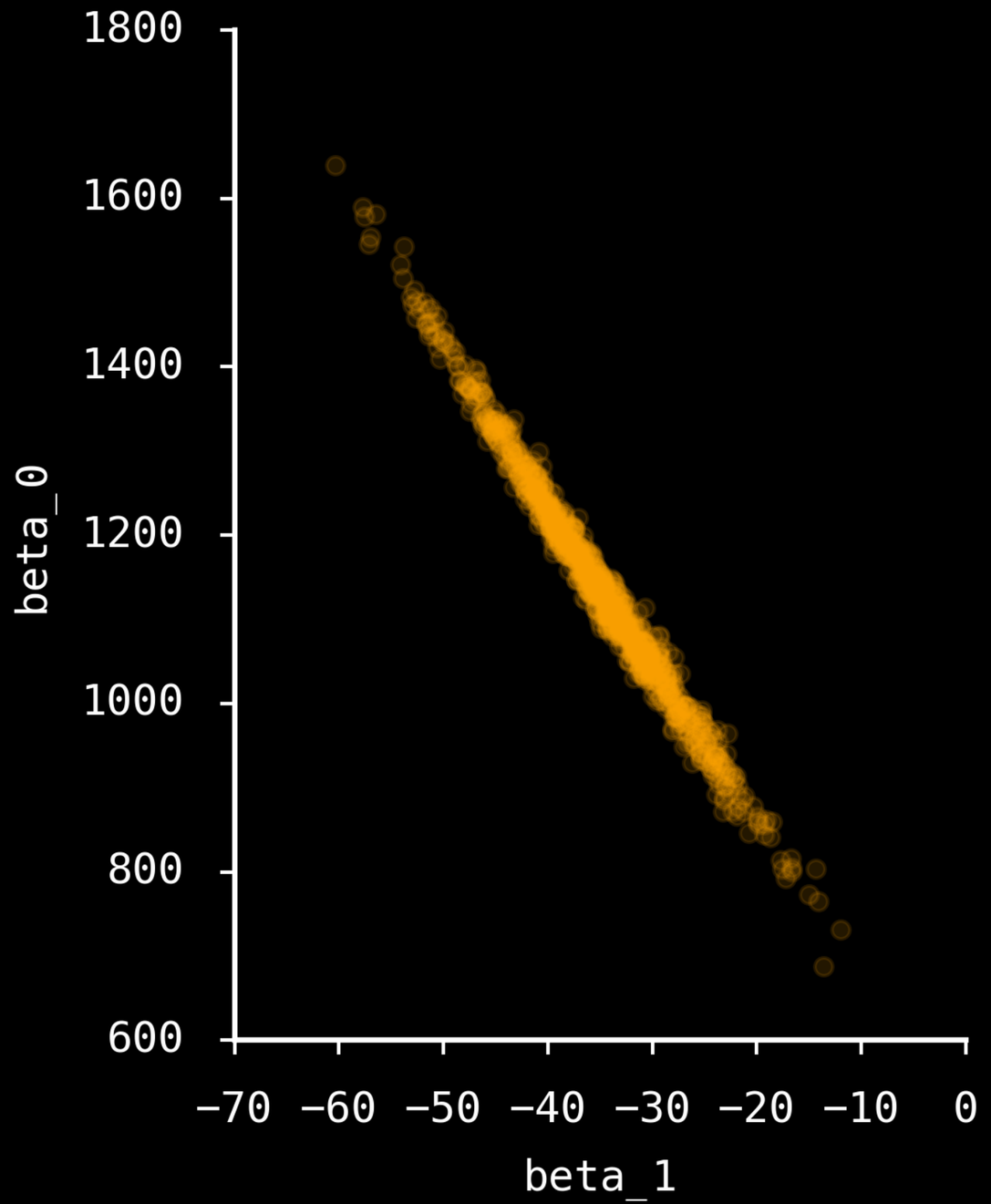
Select Model Parameters?

Need more data for the model:

Bootstrap


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl


1800
1600
1400
1200
1000
800
600
beta_0
−70
−60
−50
−40
−30
−20
−10
0
beta_1

Model-Vis Approach

---

[3] Visualise with different input datasets

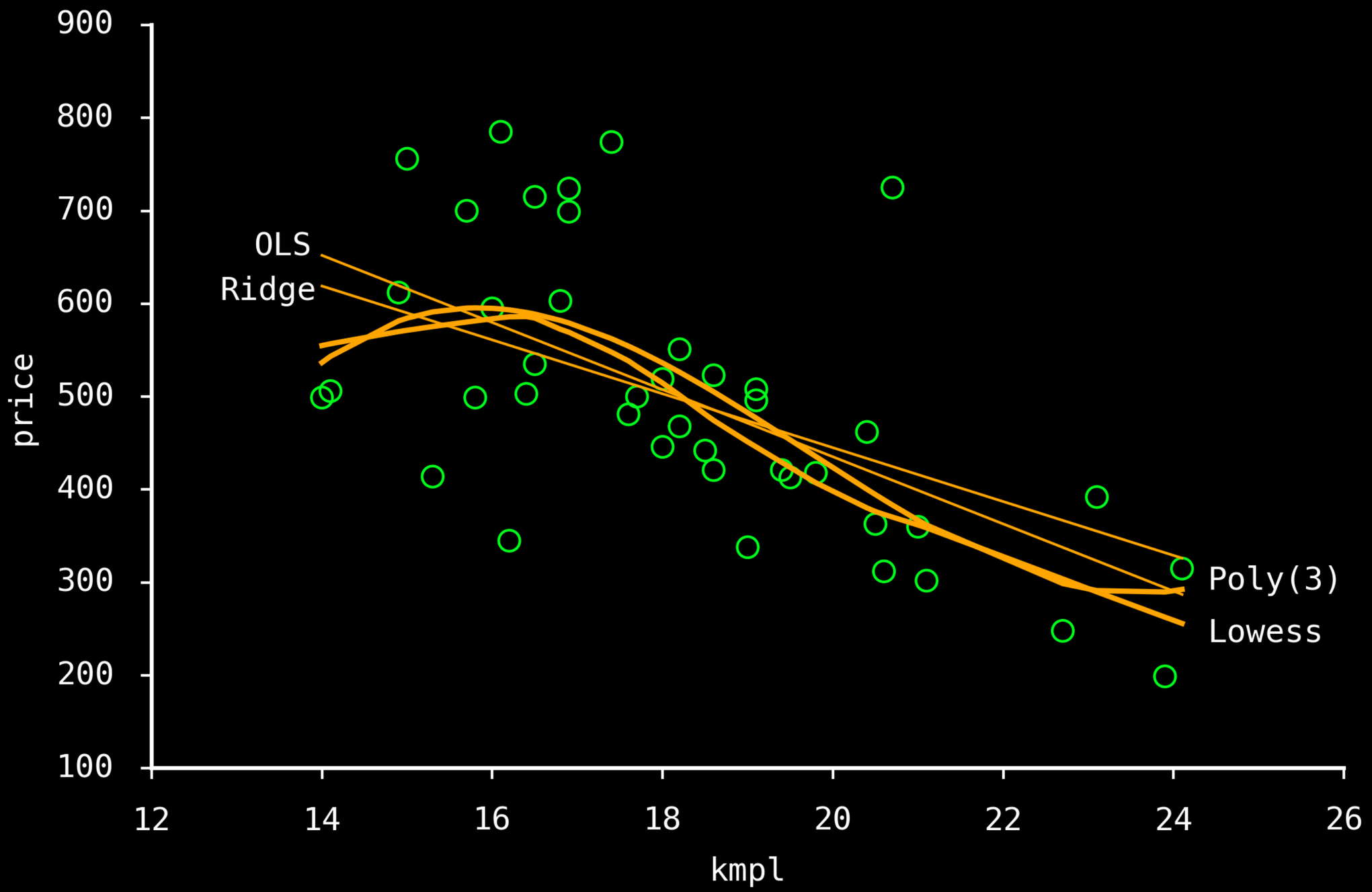
Add More Models

Ordinary Least Square (OLS)

Ridge Regression (alpha = 0.1)

Polynomial (n=3)

LOWESS


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl
OLS
Ridge
Poly(3)
Lowess

Show the Model Space

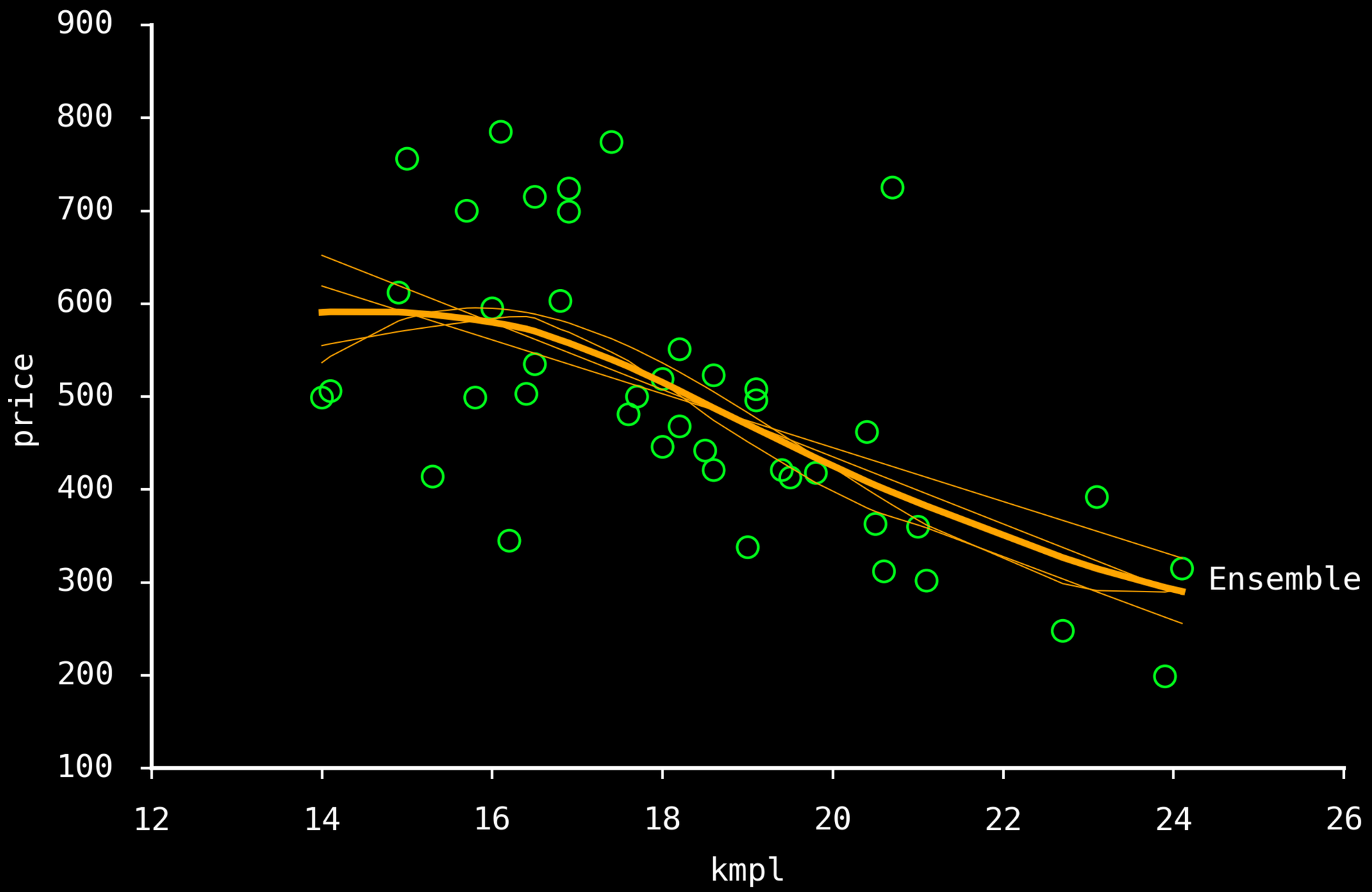
Ordinary Least Square (OLS)

Ridge Regression (alpha = 0.1)

Polynomial (n=3)

LOWESS

ENSEMBLE (Averaging)


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl
Ensemble

Model-Vis Approach

---

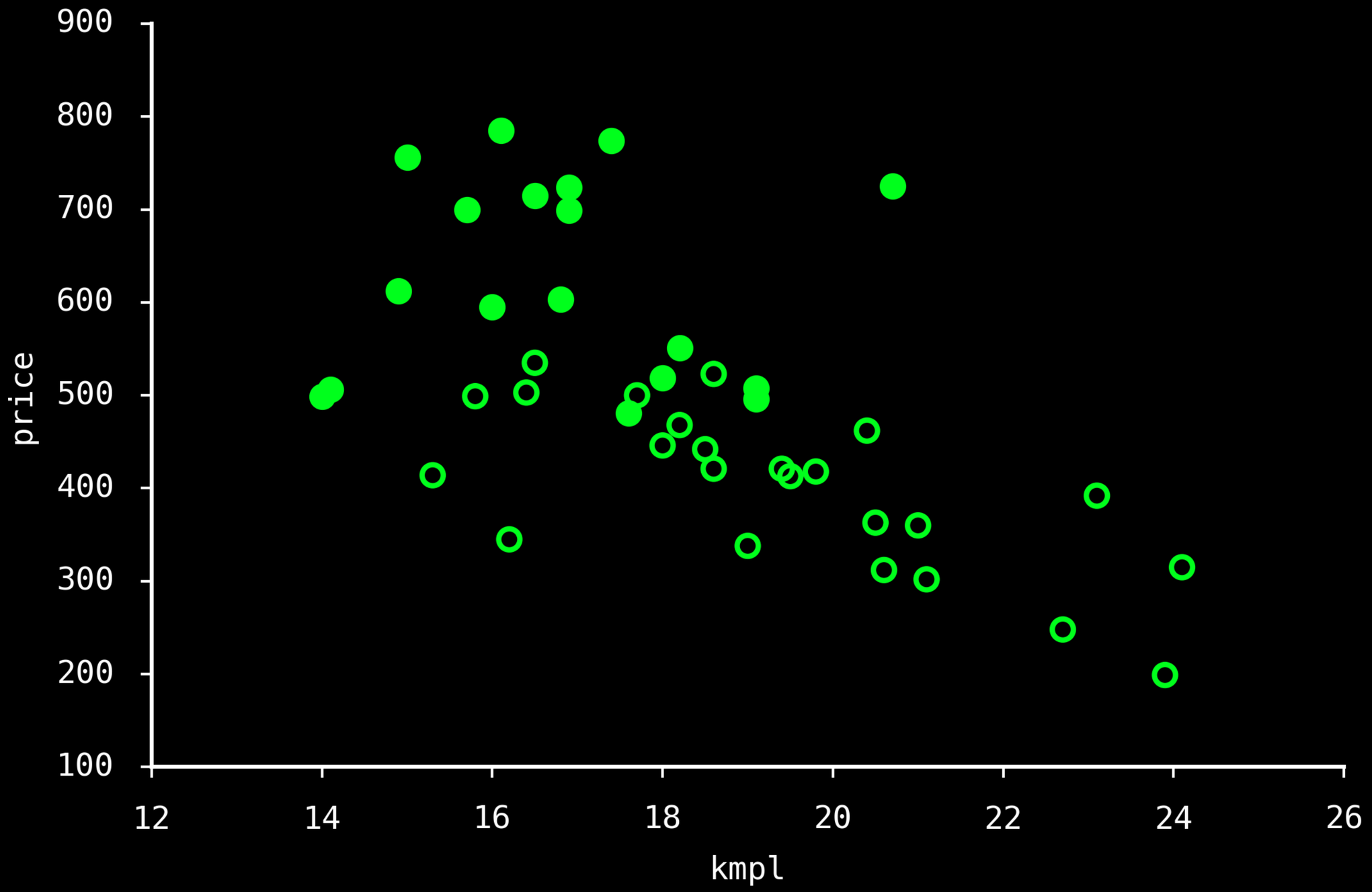
[4] Visualise the model space

Add More Features?

Search for a better explanation between price and kmpl

type: hatchback, sedan

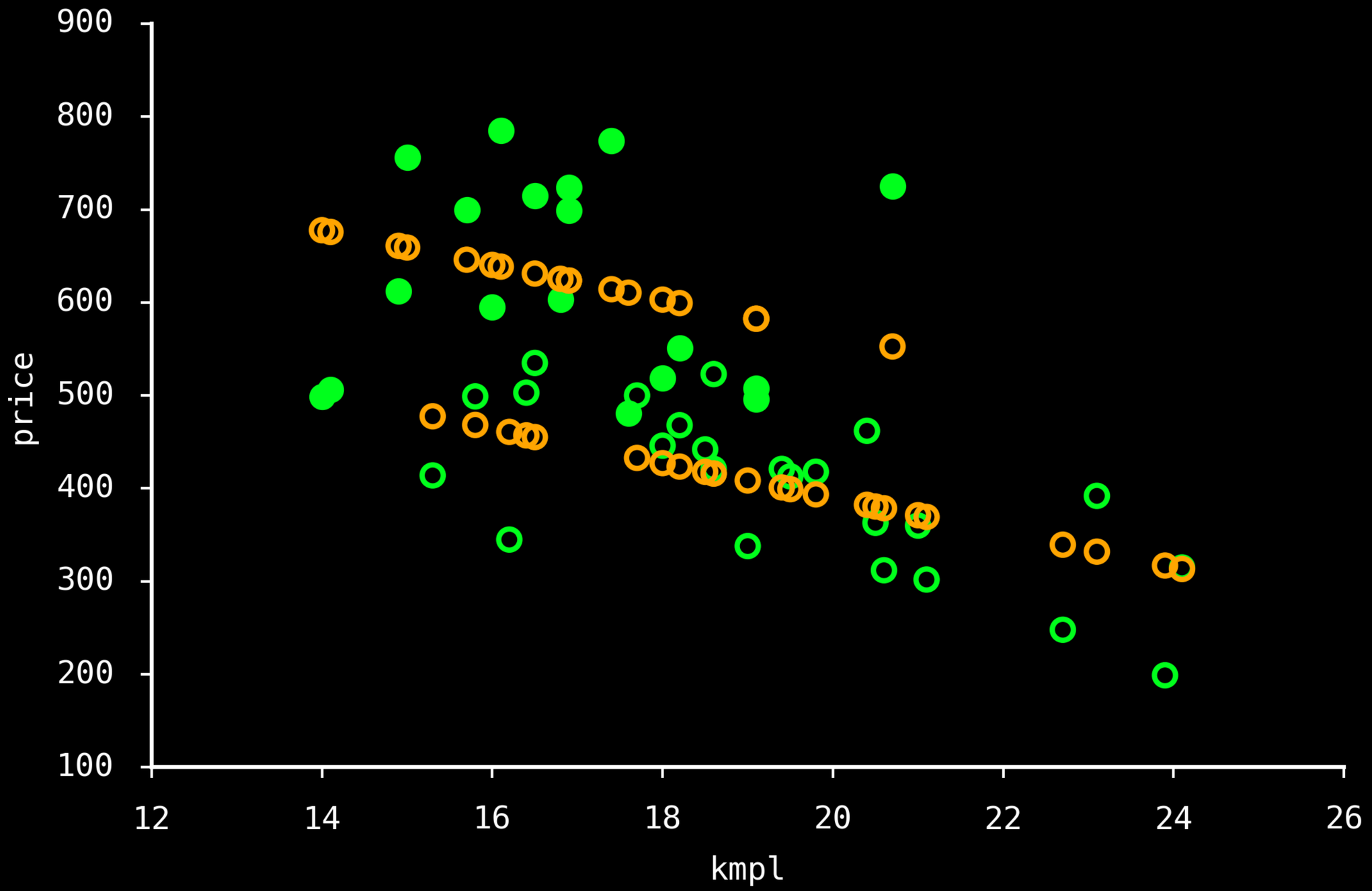
brand	model	price	kmpl	type
Tata	Nano	199	23.9	Hatchback
Suzuki	Alto800	248	22.7	Hatchback
Hyundai	EON	302	21.1	Hatchback
Nissan	Datsun	312	20.6	Hatchback
...	...	...	...	...
Suzuki	Ciaz	725	20.7	Sedan
Skoda	Rapid	756	15.0	Sedan
Hyundai	Verna	774	17.4	Sedan
VW	Vento	785	16.1	Sedan


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Add Features to One Model

OLS Regression

$$price = \beta_0 + \beta_1 * kmpl + \beta_2 * type$$


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Build Two Models

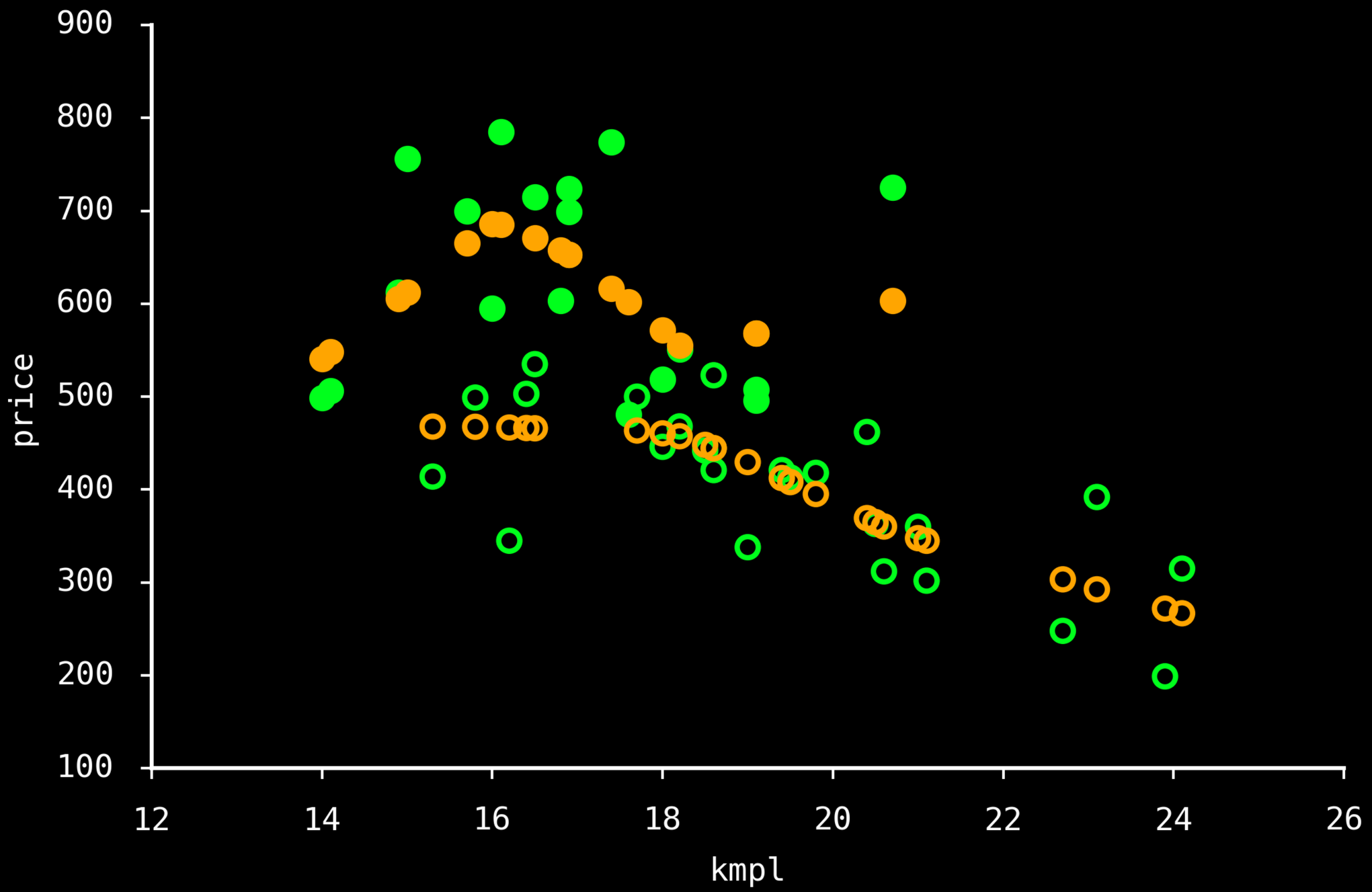
---

type = hatchback:

$$price_h = lowess(kmpl_h)$$

type = sedan:

$$price_s = lowess(kmpl_s)$$


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Model-Vis Approach

---

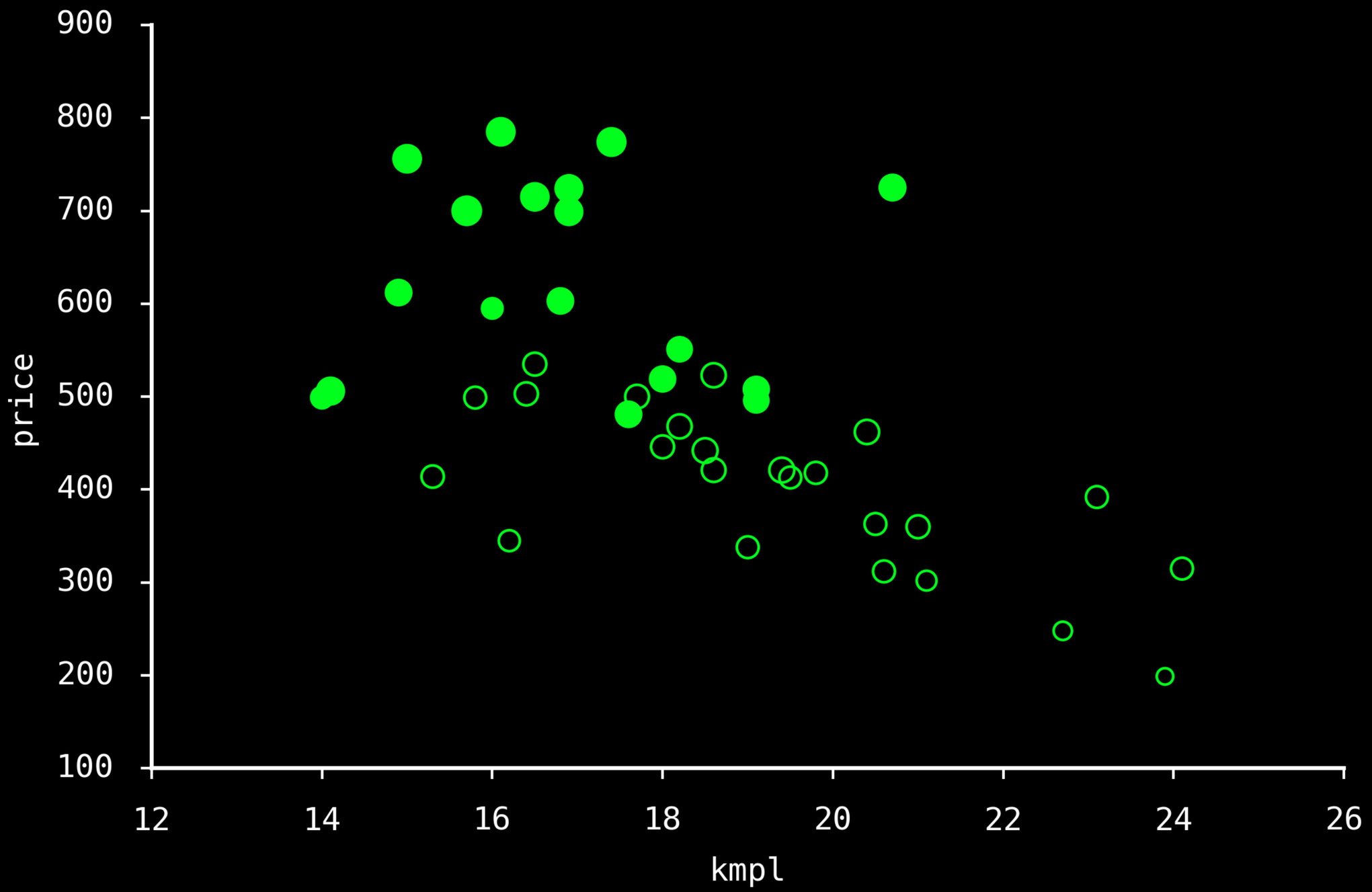
[5] Visualise the many models together

Keep Adding Feature

Can get complex really fast!

Add one more feature: bhp

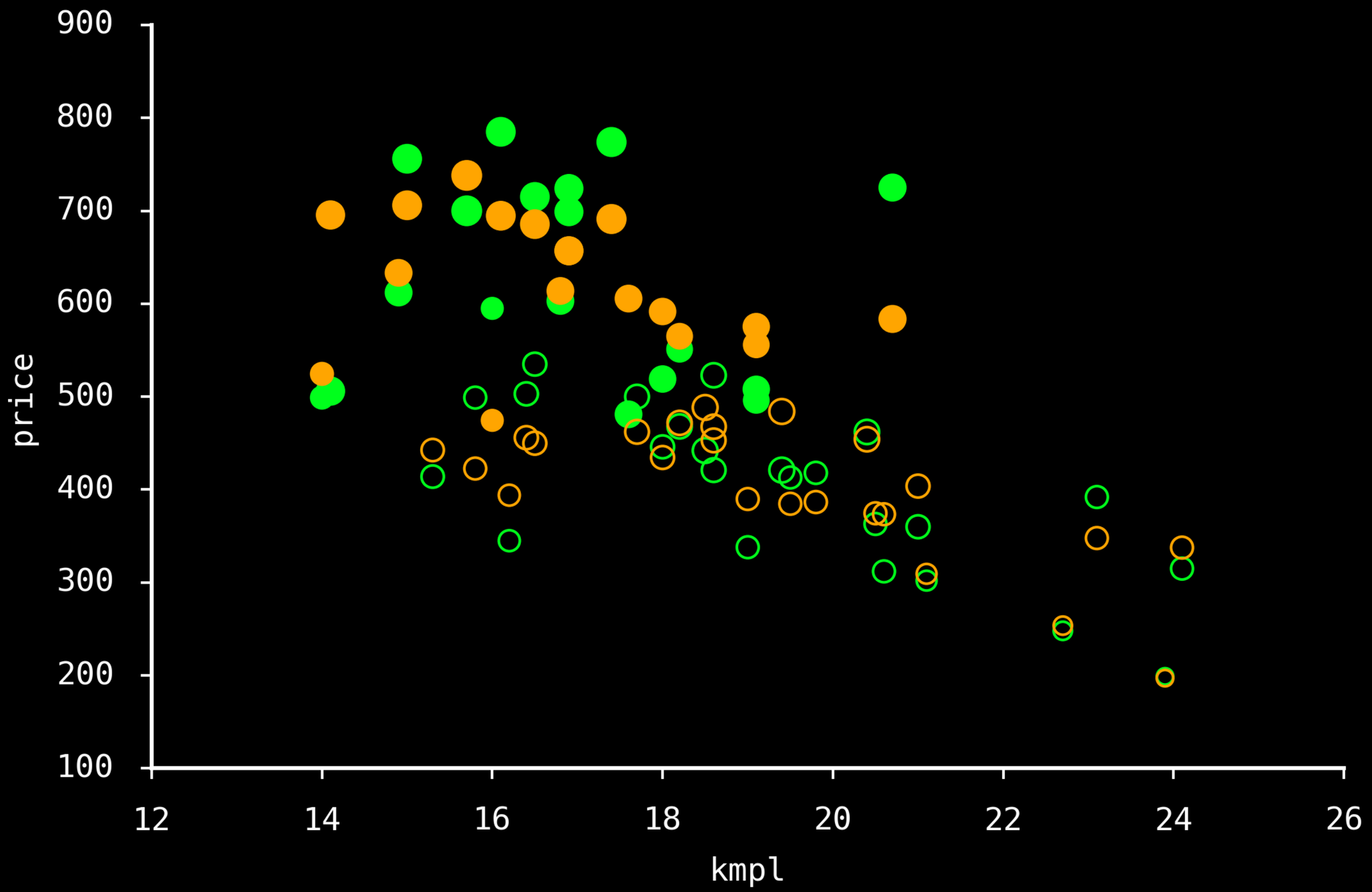
brand	model	price	kmpl	type	bhp
Tata	Nano	199	23.9	Hatchback	38
Suzuki	Alto800	248	22.7	Hatchback	47
Hyundai	EON	302	21.1	Hatchback	55
Nissan	Datsun	312	20.6	Hatchback	67
...	...	...	...	...	...
Suzuki	Ciaz	725	20.7	Sedan	91
Skoda	Rapid	756	15.0	Sedan	104
Hyundai	Verna	774	17.4	Sedan	106
VW	Vento	785	16.1	Sedan	104


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Build a New Model

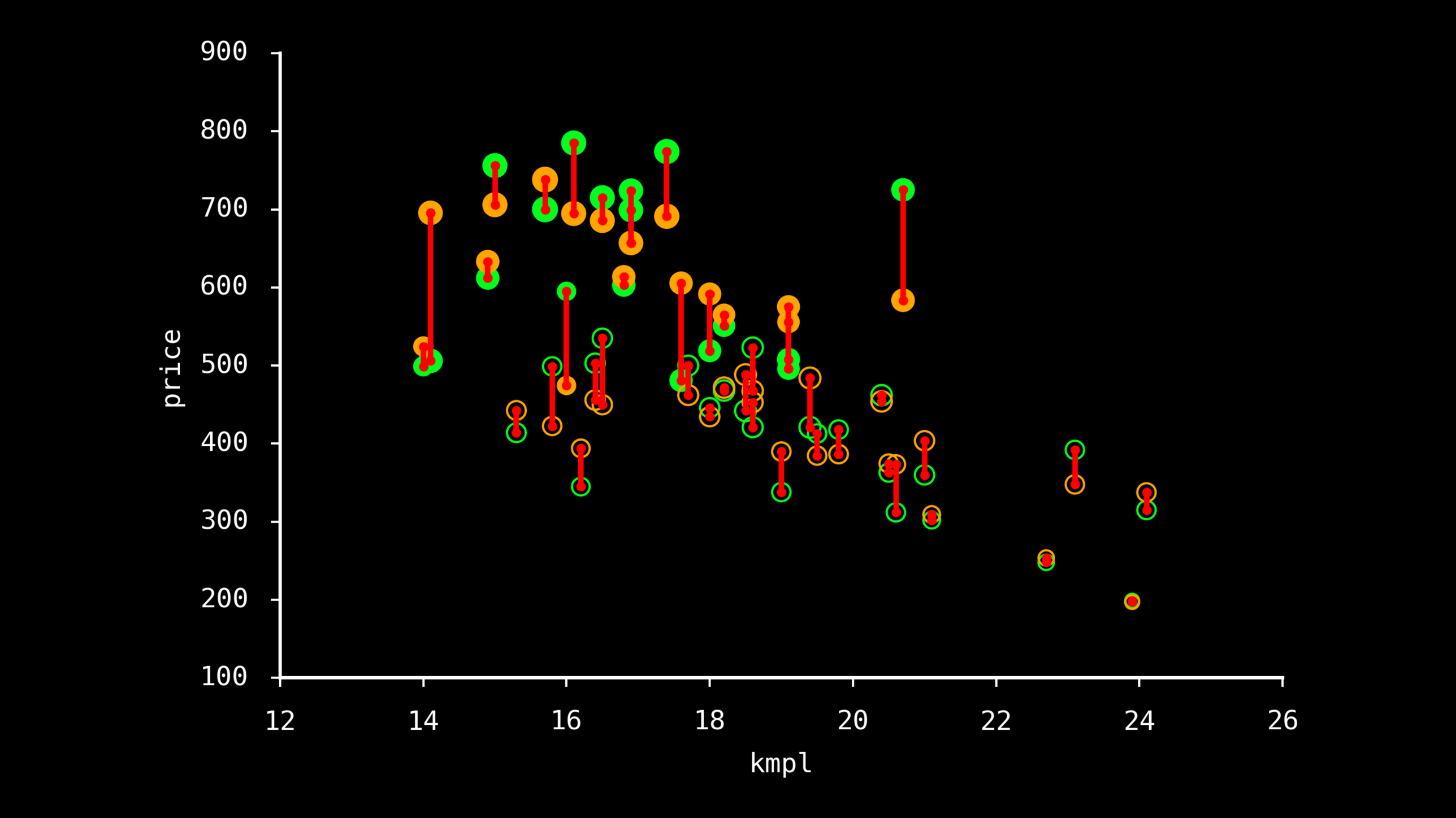
OLS Regression

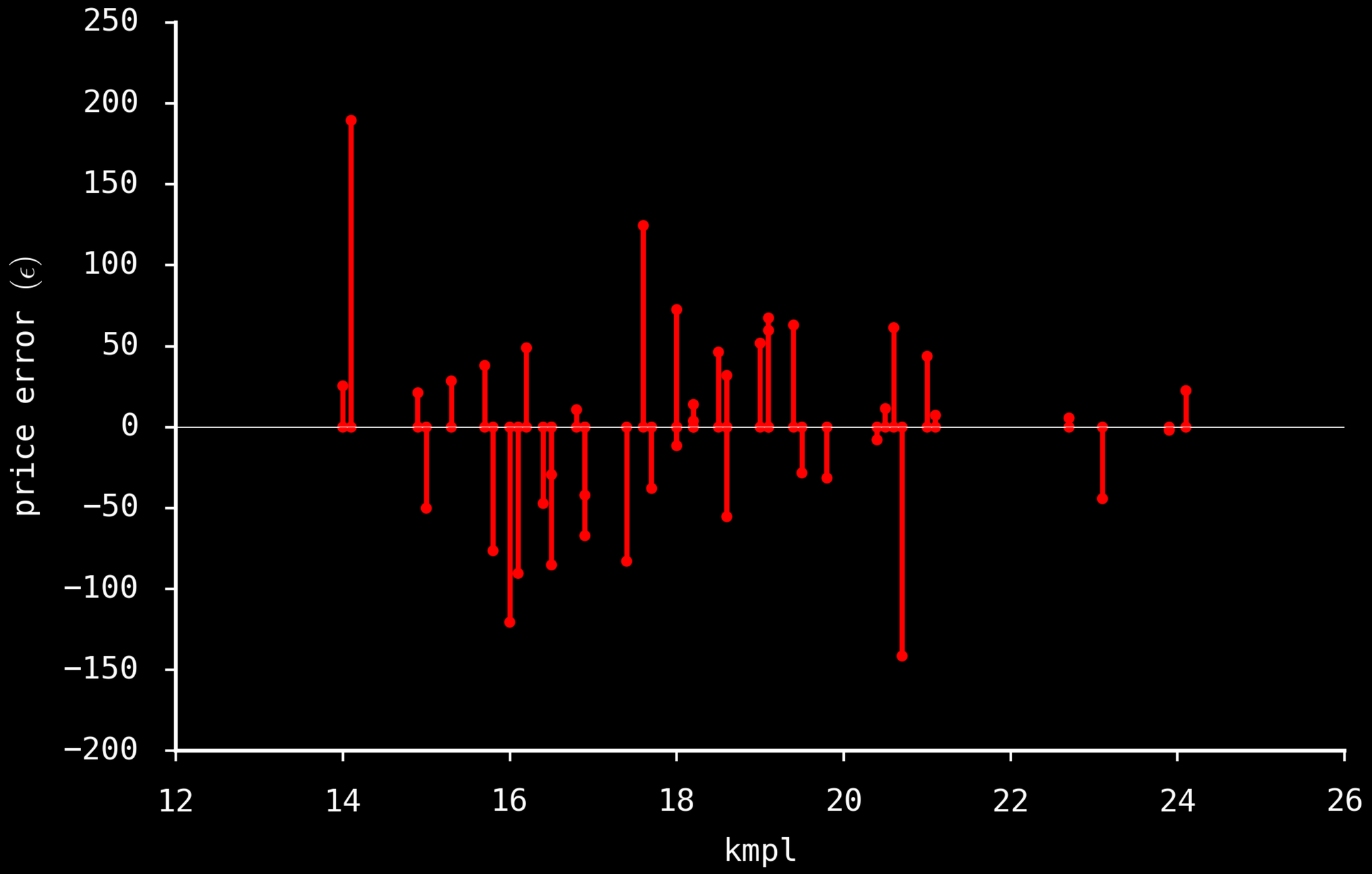
$$price = \beta_0 + \beta_1 * kmpl + \beta_2 * type + \beta_3 * bhp$$


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Visualise Model Errors

Visualise $\epsilon = price_{pred} - price$


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl


250
200
150
100
50
0
−50
−100
−150
−200
price error (€)
12
14
16
18
20
22
24
26
kmpl

Errors Easier to Visualise

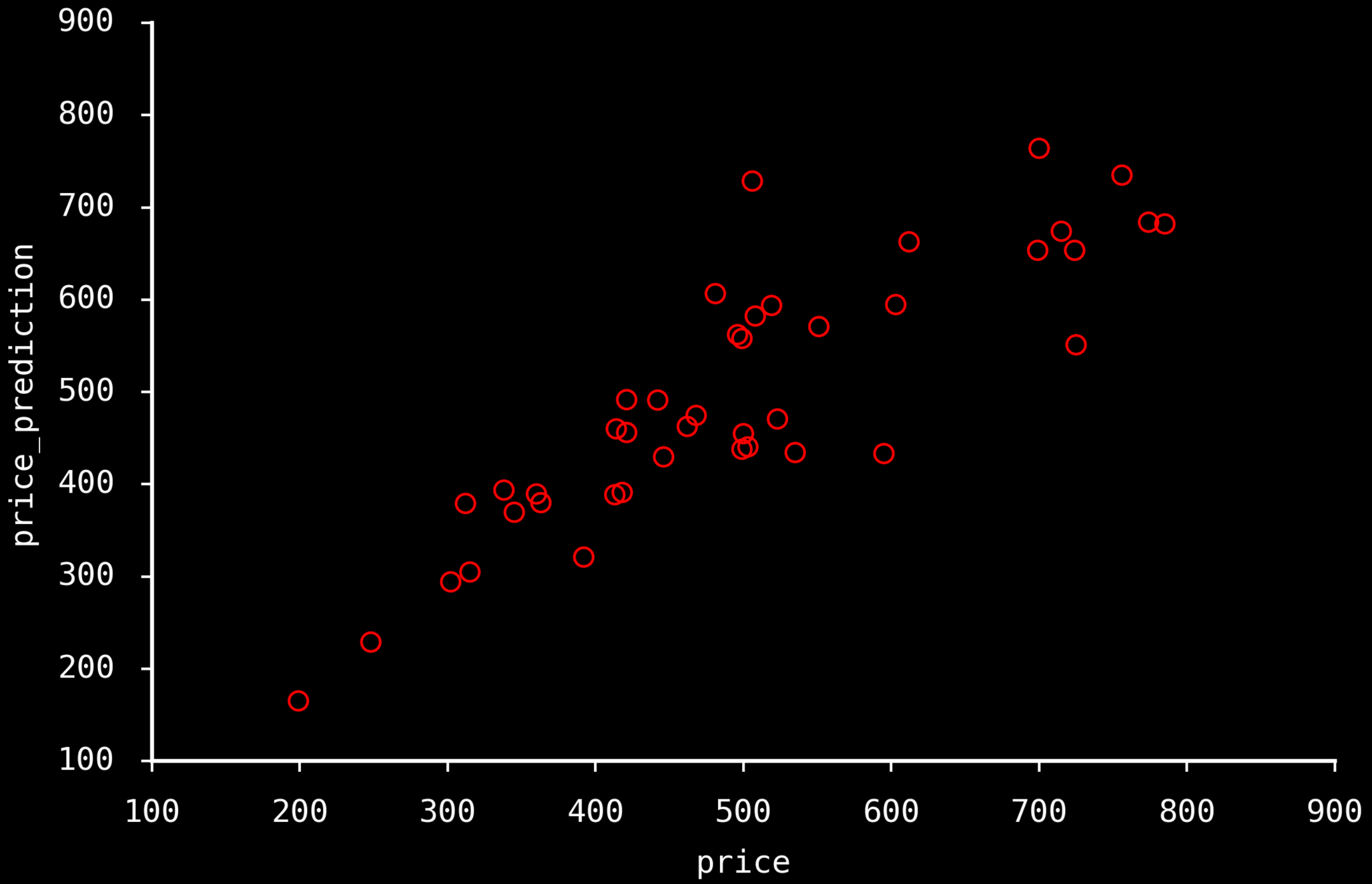
Move away from features

Address curse of dimensionality

Prediction vs. Error

Test Robustness of the Model

Do Cross Validation - 6 fold

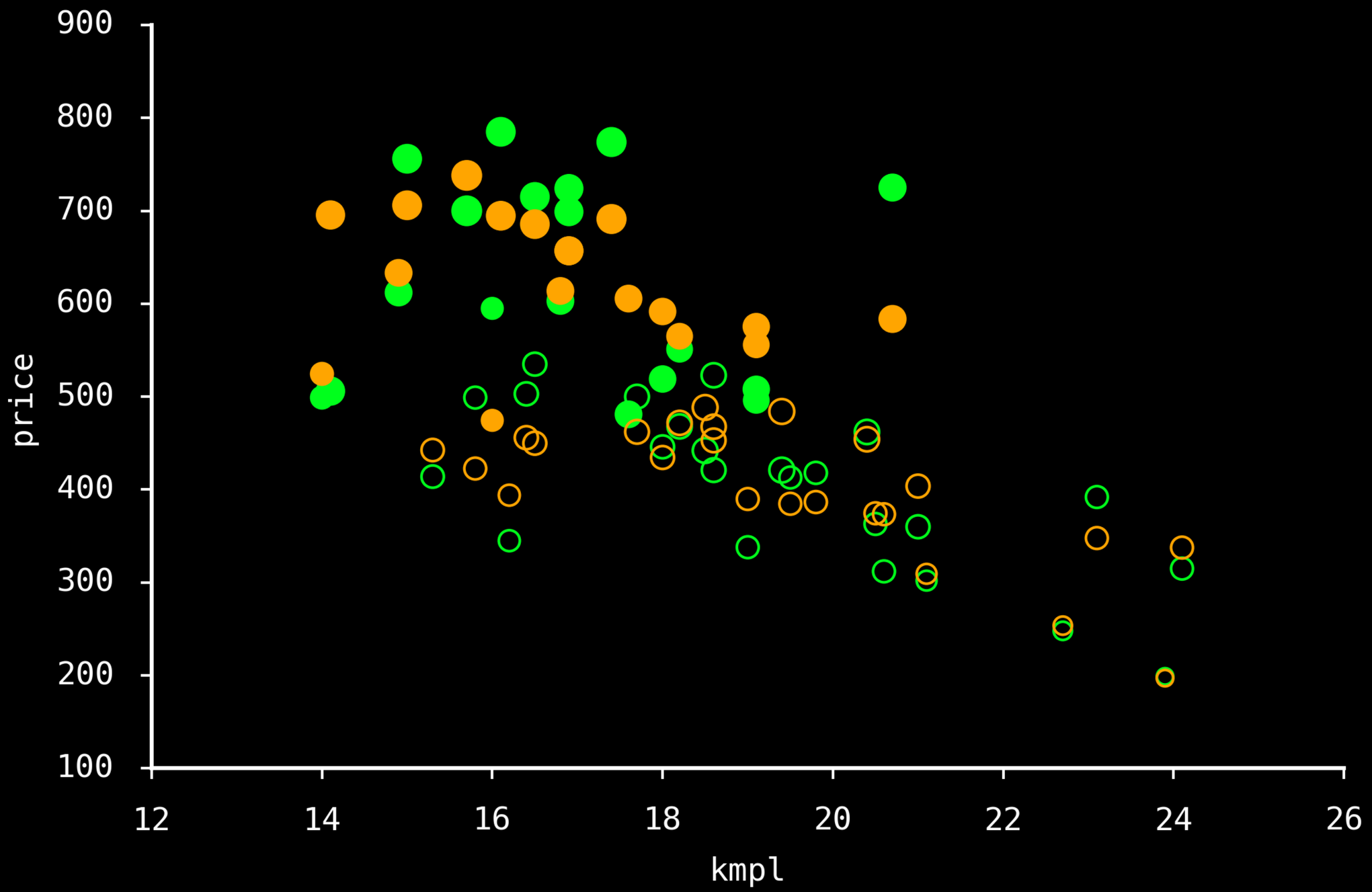

900
800
700
600
500
400
300
200
100
price_prediction
100
200
300
400
500
600
700
800
900
price

Build the final Model

OLS Regression

$$price = \beta_0 + \beta_1 * kmpl + \beta_2 * type + \beta_3 * bhp$$

Cross Validation - 6 fold


900
800
700
600
500
400
300
200
100
price
12
14
16
18
20
22
24
26
kmpl

Model-Vis Approach

---

[6] Visualise the errors in model fitting

Model-Vis Approach

---

[0] Visualise the data space
[1] Visualise the predictions in the data space
[2] Visualise with different model parameters
[3] Visualise with different input datasets
[4] Visualise the entire model space
[5] Visualise the many models together
[6] Visualise the errors in model fitting

Model-Vis & ML Approach

---

[0] **DATA VIS**: the data space
[1] **PREDICTION**: the predictions in the data space
[2] **TUNING**: with different model parameters
[3] **BOOTSTRAP**: with different input datasets
[4] **ENSEMBLE**: the entire model space
[5] **N-MODELS**: the many models together
[6] **VALIDATION**: the errors in model fitting

Model Explosion

Base Models = 7

OLS Regression (p = 1, p = 2, p = 3)
Ridge Regression, Polynomial, LOWESS (total, by type)

+ Add Tuning Models
+ Add Bootstrap Models
+ Add Ensemble Models
+ Add Cross-Validation Models

Challenge with Model

Keep track of prediction & errors

Keep track of model output parameters

Tidy Model

Augment predictions & errors to dataset

Create output parameters data frame

Visualise like Tidy Data

Tooling

Started with python and challenging **broom** package in R (by David Robinson)

```r
library(dplyr)
library(broom)
set.seed(2014)

# Move Bootstrap Output to Tidy Model
bootcars <- cars %>%
            bootstrap(100) %>%
            do(augment(smooth.spline(.$kmpl, .$price, df=4), .))

# Plot the Output
ggplot(bootcars, aes(kmpl, price)) + geom_point() +
    geom_line(aes(y=.fitted, group=replicate), alpha=.2)
```

Model-Vis

Similar challenges to Data-Vis

More an Art, than a Science

Essential in ML Model Pipeline

Both to Explain or to Predict

Scope for easier tooling

Model-Vis

Cars Dataset (n = 833, p = 63)
https://github.com/amitkaps/cars

Talk Code
https://github.com/amitkaps/modelvis

Model Visualisation

Amit Kapoor
@amitkaps

amitkaps.com