Aprendizaje Estadístico Supervisado

Natalia da Silva

2024

Esquema

- Interpretabilidad en aprendizaje estadístico
- Gráfico de dependencia parcial (PDP)
- Extensiones del PDP (ICE y ALE)

Clase basada en: Interpretable Machine Learning book

Model agnostic

Principal ventaja de los métodos “Model agnostic” sobre los “Model specific” es su gran flexibilidad para ser usados en todos los ML.

Deseable para los métodos “Model agonostic”:

- Flexibilidad del modelo, el método funciona con cualquier ML
- Flexibilidad de la explicación, no limitado a cierta forma de la explicación
- Flexibilidad en la representación, el sistema de explicación debería ser capáz de usar distintas representaciones de las variables explicativas

Gráfico de dependencia parcial (PDP)

- Muestra el efecto marginal de una o dos variables explicativas en el valor predicho del ML.
- Muestra si la relación entre la respuesta y la variable explicativa es lineal, mónotona o más compleja. Cuando se aplica a un modelo de regresión lineal, el PDP siempre muestra una relación lineal.

Gráfico de dependencia parcial (PDP)

Separamos los predictores en dos grupos:

- X_S: la o las variables explicativas cuyo efecto sobre la respuesta queremos describir
- x_C son las otras variables explicativas utilizadas en el modelo

La función de dependencia parcial para regresión es:

$$f_S(x_S) = E_{X_C}[\hat{f}(x_S, x_C)] = \int \hat{f}(x_S, x_C) dP(x_C)$$

- Marginalizando sobre x_C se obtienen una función que depende solamente de las variables en S e interacciones con otras variables incluídas.

Gráfico de dependencia parcial (PDP)

- La función de dependencia parcial $\hat{f}_{x_S}$ es estimada calculando el promedio en los datos de entrenamiento (Monte Carlo)

$$\hat{f}_{x_S}(x_S) = \frac{1}{n}\sum_{i=1}^{n} \hat{f}(x_s, x_C^{(i)})$$

- $x_C^{(i)}$ son los valores de las variables que no estamos interesados en el conjunto de datos.
- La función nos dice que para un valor determinado en las variables en S cuál es el efecto marginal promedio en las predicciones.

Gráfico de dependencia parcial (PDP)

- Un supuesto en PDP es que las variables explicativas en C no están correlacionadas con las variable en S.
- Si este supuesto es violado el promedio calculado para el PDP incluirá puntos que son muy improbables o inluso imposibles.
- Para clasificación el PDP presenta la probabilidad para cierta clase dada diferentes valores de las variables en S. Para muchas clases se puede dibujar una linea o gráficos por clase.
- Para predictoras categóricas, para cada categoría se obtiene el PDP estimado forzando todos los datos a la misma categoría.

PDP pasos

1. Selecciono una o dos variables de interés x_S
2. Definimos una grilla para x_S
3. Para cada valor de la grilla: remplazo la variable de interés con el valor de la grilla y promedio las predicciones.
4. Dibujo la curva

Ejemplo: Datos Ames

Particiono los datos

```
library(modeldata)
data(ames)
library(tidymodels)
tidymodels_prefer()

set.seed(501)

# Save the split information for an 80/20 split of the dat
ames_split <- initial_split(ames, prop = 0.80)
ames_split
```

```
<Training/Testing/Total>
<2344/586/2930>
```

```
ames_train <- training(ames_split)
ames_test  <-  testing(ames_split)
```

Ejemplo: Datos Ames

Ajusto árbol

```
tree_model <-
  decision_tree(min_n = 2) %>%
  set_engine("rpart") %>%
  set_mode("regression")

tree_fit <-
  tree_model %>%
  fit(Sale_Price ~   Neighborhood + Gr_Liv_Area + Year_Bui
```

Ejemplo: Datos Ames

Bosque

```
rf_model <-
  rand_forest(trees = 1000) %>%
  set_engine("randomForest") %>%
  set_mode("regression")

rf_wflow <-
  workflow() %>%
  add_formula(
    Sale_Price ~ Neighborhood + Gr_Liv_Area + Year_Built +
      Latitude + Longitude) %>%
  add_model(rf_model)

rf_fit <- rf_wflow %>% fit(data = ames_train)
```

Ejemplo: Datos Ames

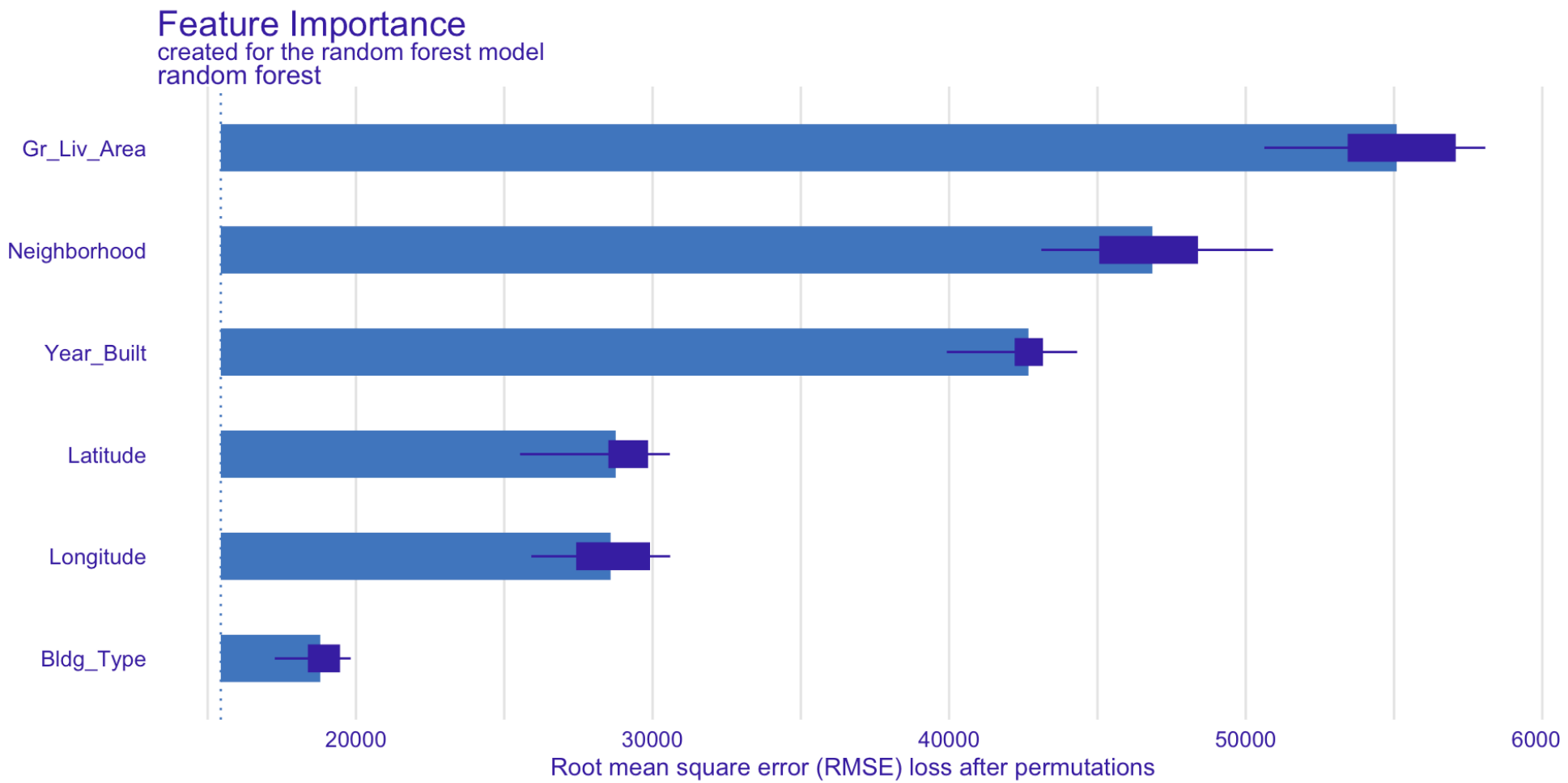
Importancia permutada, seleccionamos algunas variables para que no demore.

```
library(DALEXtra)

vip_features <- c("Neighborhood", "Gr_Liv_Area", "Year_Bui
                  "Bldg_Type", "Latitude", "Longitude")

vip_train <-
  ames_train %>%
  select(all_of(vip_features))

#explain_tidymodels crea un explainer para el workflow de

explainer_rf <-
  explain_tidymodels(
   model= rf_fit,
    data = vip_train
```

Ejemplo: Datos Ames

Importancia permutada

```
1  plot(vip_rf)
```



Ejemplo: Datos Ames

¿ Cómo cambia si lo hago para el árbol?

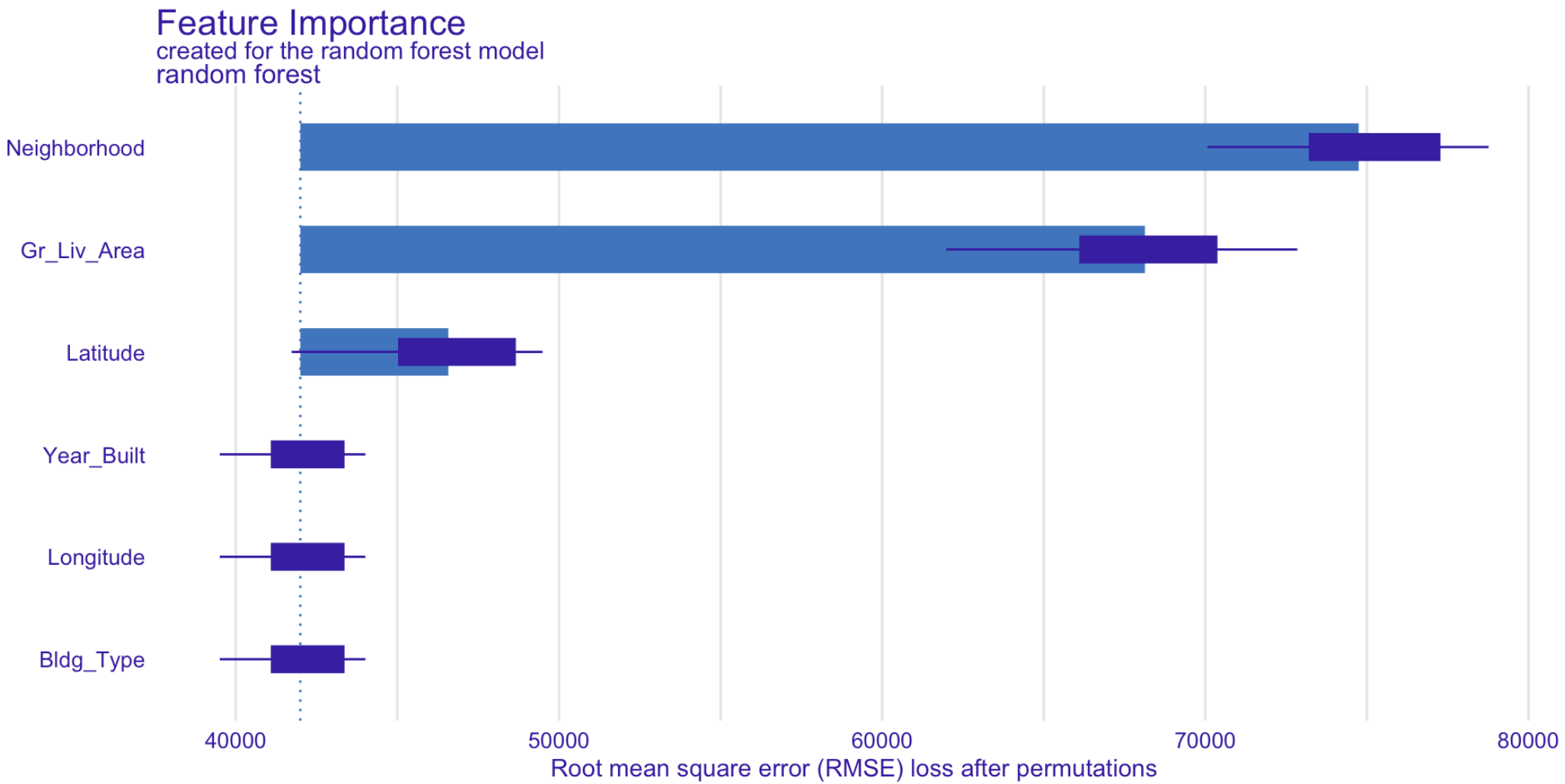
Ejemplo: Datos Ames

¿ Cómo cambia si lo hago para el árbol?

```
explainer_tree <-
  explain_tidymodels(
   model= tree_fit,
    data = vip_train,
    y = ames_train$Sale_Price,
    label = "random forest",
    verbose = FALSE
  )
set.seed(1804)
vip_tree <- model_parts(explainer_tree, loss_function = lo
```

Ejemplo: Datos Ames

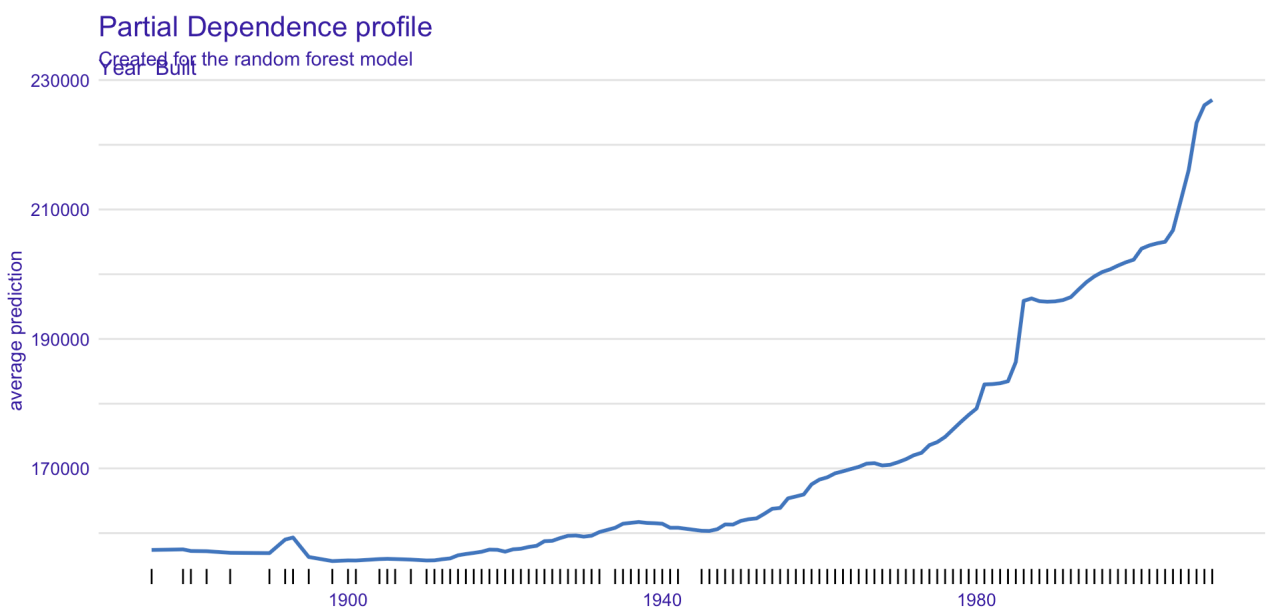
```
1  plot(vip_tree)
```



Ejemplo: Datos Ames

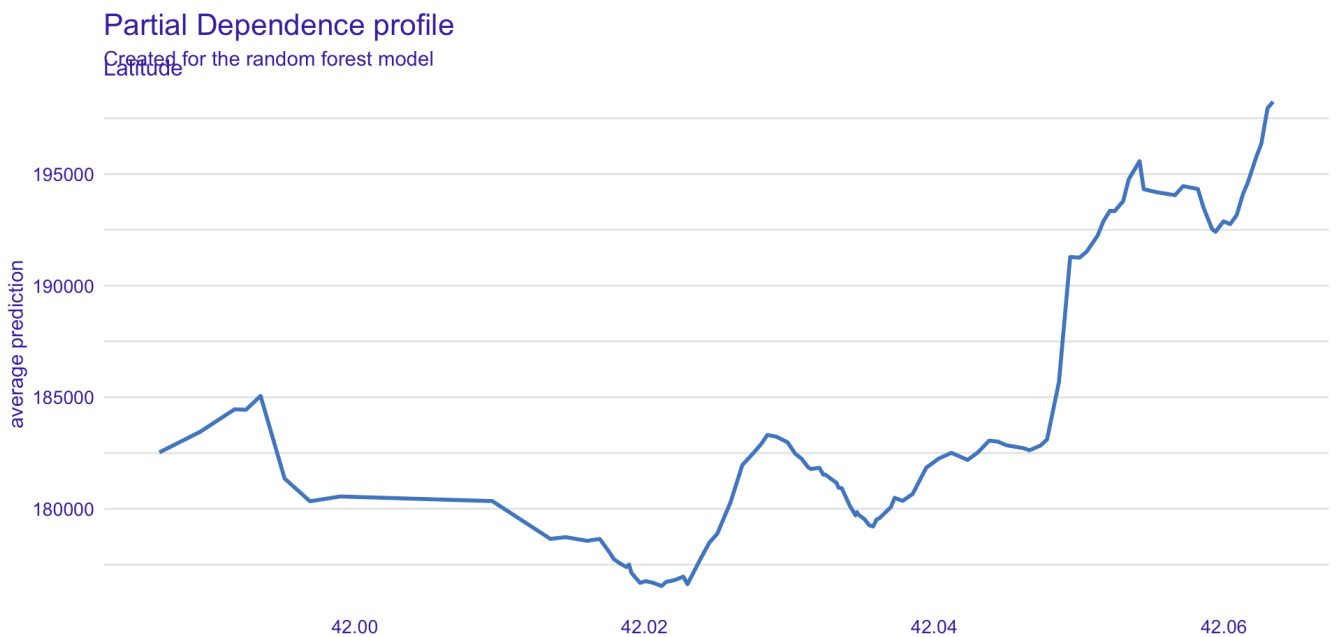
Gráfico de Dependencia Parcial (PDP)

```
1 set.seed(1805)
2 pdp_age <- model_profile(explainer_rf, N = 500, variables
3
4
5 plot(pdp_age)+
6   geom_rug()
```



Ejemplo: Datos Ames

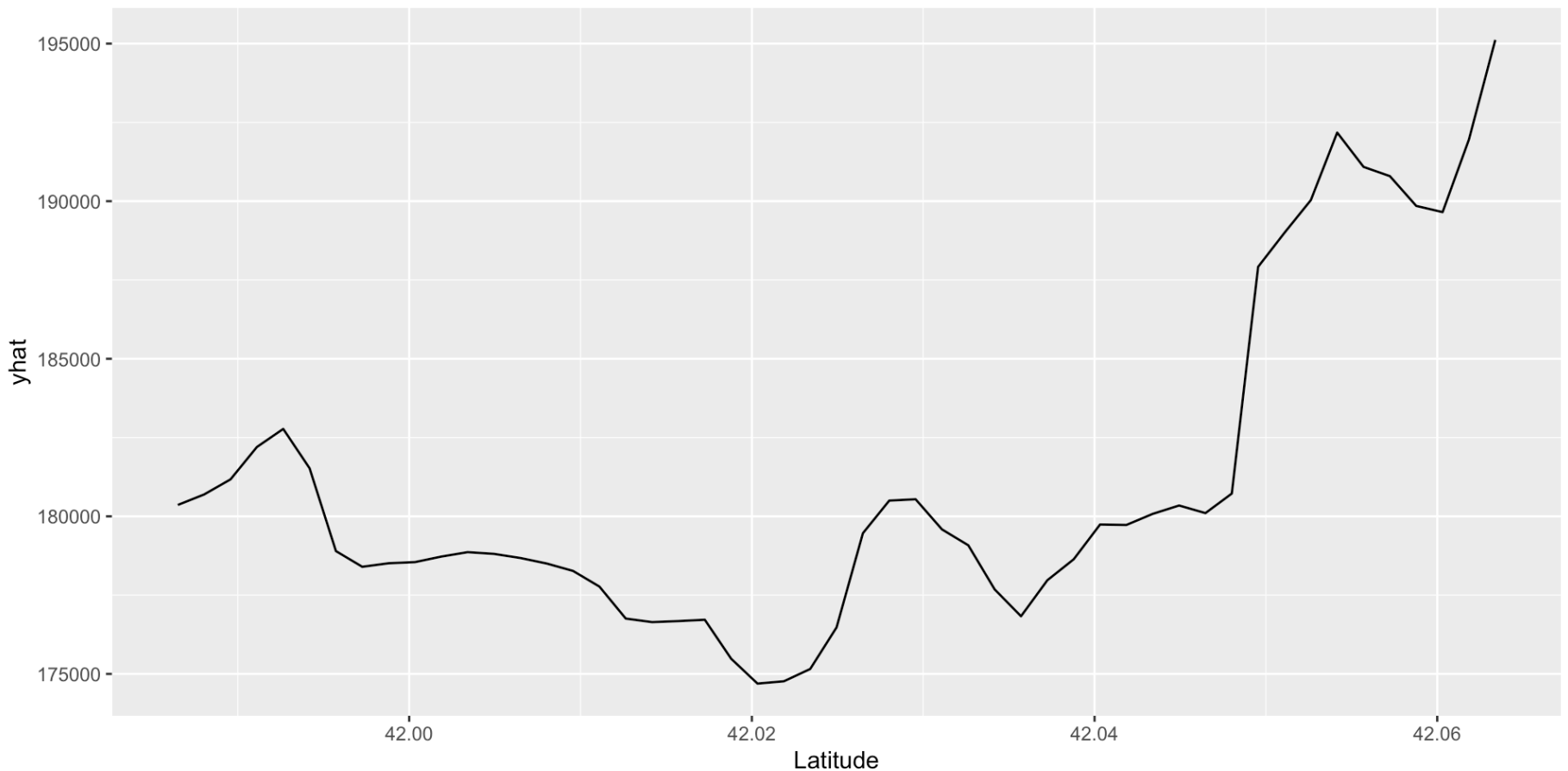
```r
set.seed(1805)
pdp_lat <- model_profile(explainer_rf, N = 500,
                                 variables = "Latitude",
                                 type='partial')


plot(pdp_lat)
```



Ejemplo: Datos Ames

Alternativamente se puede usar el paquete pdp

```
1 library(pdp)
2 pdp::partial(extract_fit_parsnip(rf_fit), pred.var = "Lati
```



Desventaja

- El máximo número de variables en un PDP con sentido es 2.
- Algunos PDP no muestran la distribución de x_C en los datos, problema porque puedo sobre interpretar los resultados en lugares donde no observó datos o muy pocos.
- El supuesto de independencia es el principal problema en PDP, x_S no está correlacionada con otras x_C
- Efectos de heterogenidad pueden estar ocultos porque los PDP solo muestran el efecto marginal promedio.

Esperanza condicional individual (ICE)

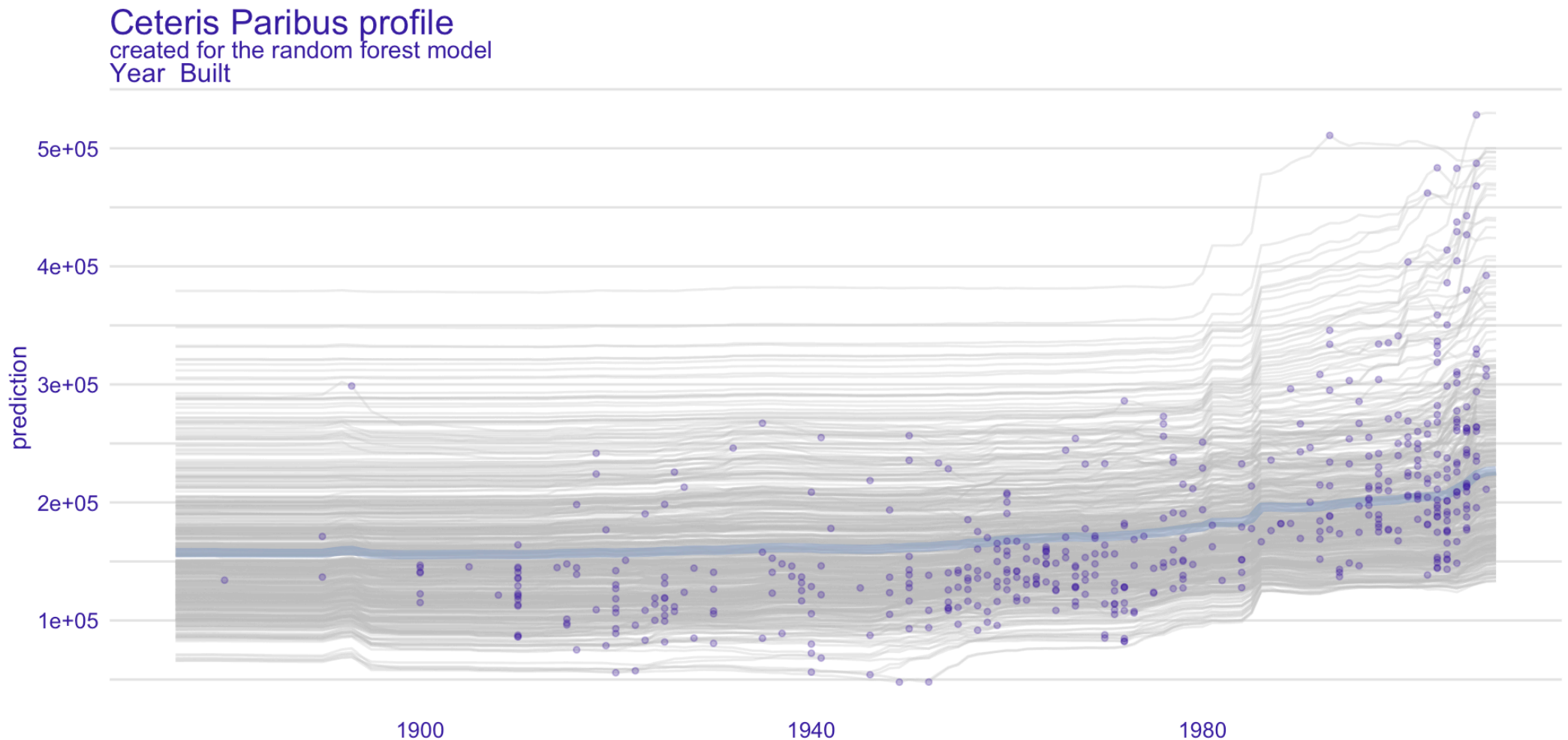
- ICE muestra una linea por observación, muestra cómo cambia la predicción cuando cambia una observación

Para cada observación en $\{(x_S^{(i)}, x_C^{(i)})\}_{i=1}^{N}$ la curva $f_S^{(i)}$ es dibujada contra $x_S^{(i)}$ mientras $x_C^{(i)}$ permanece constante.

- ICE permite visualizar la dependencia en la predicción de una variable para cada observación separadamente
- PDP es el promedio de las lineas del ICE.
- En el caso que hay interacción entre x_C y x_S es mejor que el PDP.

Ejemplo: Datos Ames

```
1  plot(pdp_age,  geom = "points", variables = "Year_Built",
```



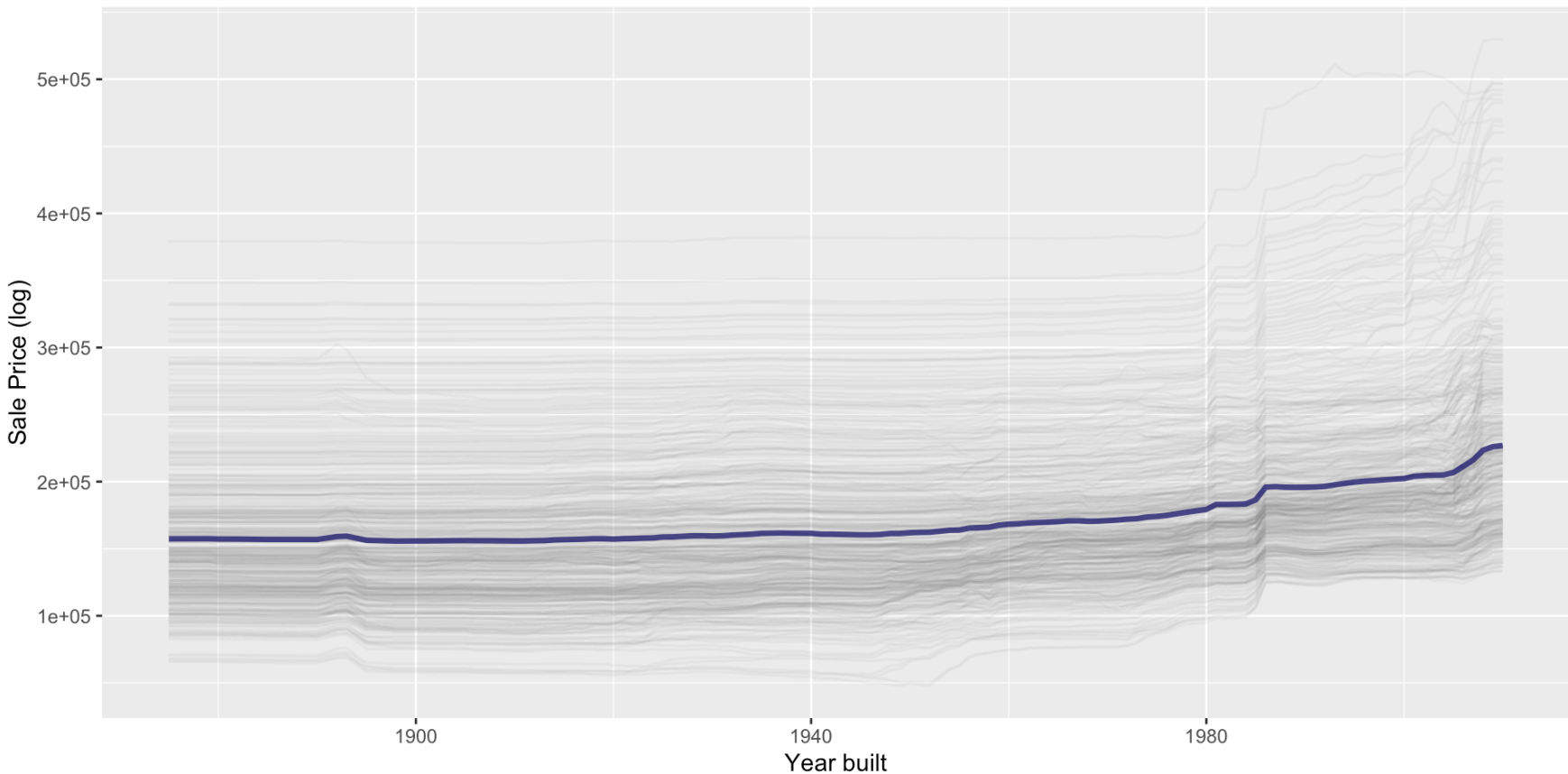
Ejemplo: Datos Ames

Alternativamente

```
ggplot_pdp <- function(obj, x) {

  p <-
    as_tibble(obj$agr_profiles) %>%
    mutate(`_label_` = stringr::str_remove(`_label_`, "^[^
    ggplot(aes(`_x_`, `_yhat_`)) +
    geom_line(data = as_tibble(obj$cp_profiles),
              aes(x = {{ x }}, group = `_ids_`),
              linewidth = 0.5, alpha = 0.05, color = "gray

  num_colors <- n_distinct(obj$agr_profiles$`_label_`)

  if (num_colors > 1) {
    p <- p + geom_line(aes(color = `_label_`), linewidth =
  } else {
```

Ejemplo: Datos Ames

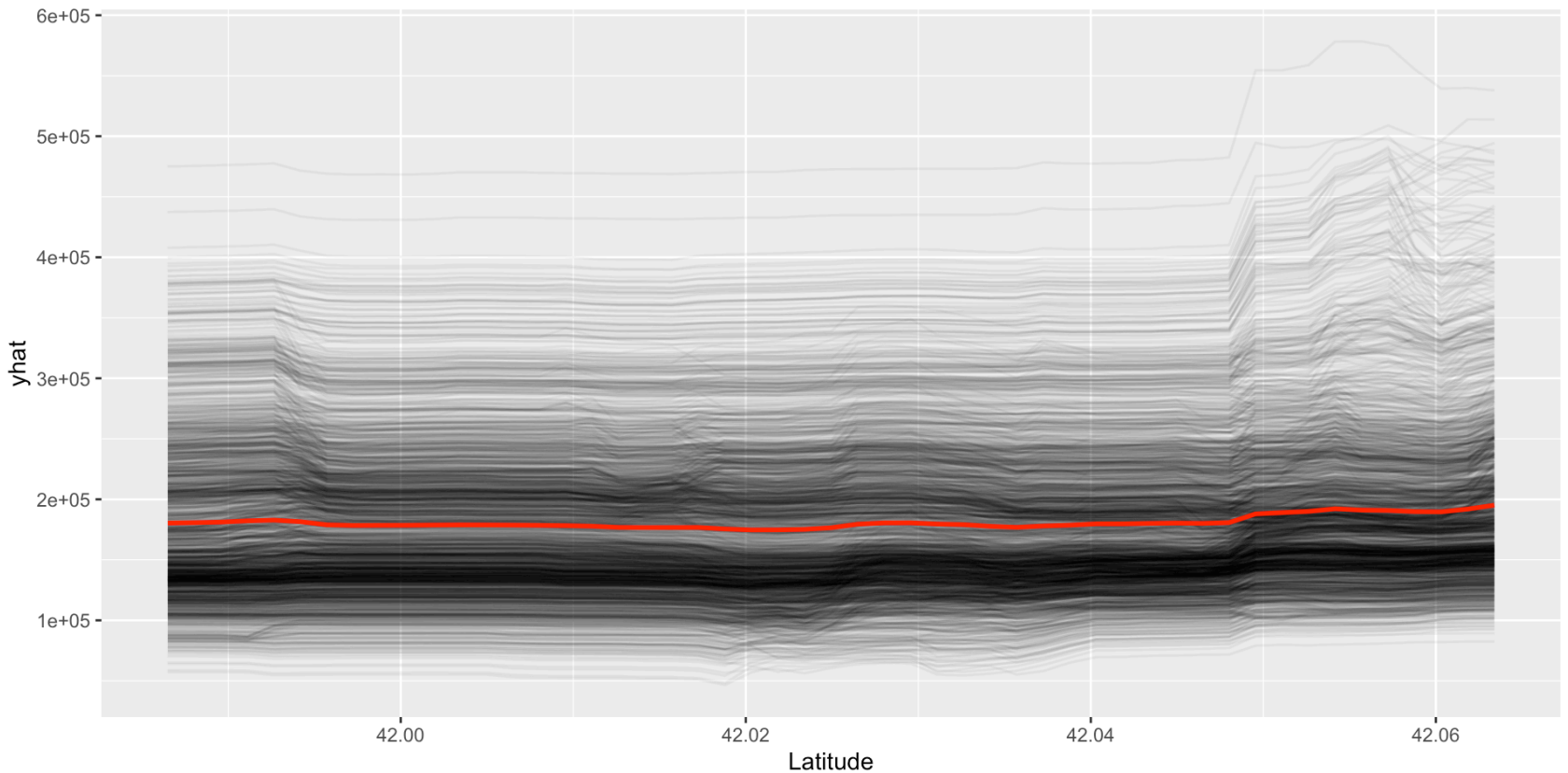
```
ggplot_pdp(pdp_age, Year_Built)  +
  labs(x = "Year built",
       y = "Sale Price (log)",
       color = NULL)
```



Ejemplo: Datos Ames

Alternativamente

```
pdp::partial(extract_fit_parsnip(rf_fit), pred.var = "Lati
                 ice=T, alpha=0.05,
                plot = TRUE,prob=T,plot.engine = "ggplot2", t
```



ICE Ventajas-Desventajas

Ventaja:

1. Más intuitivos que PDP.
2. Puede descubrir relaciones heterogeneas

Desventajas:

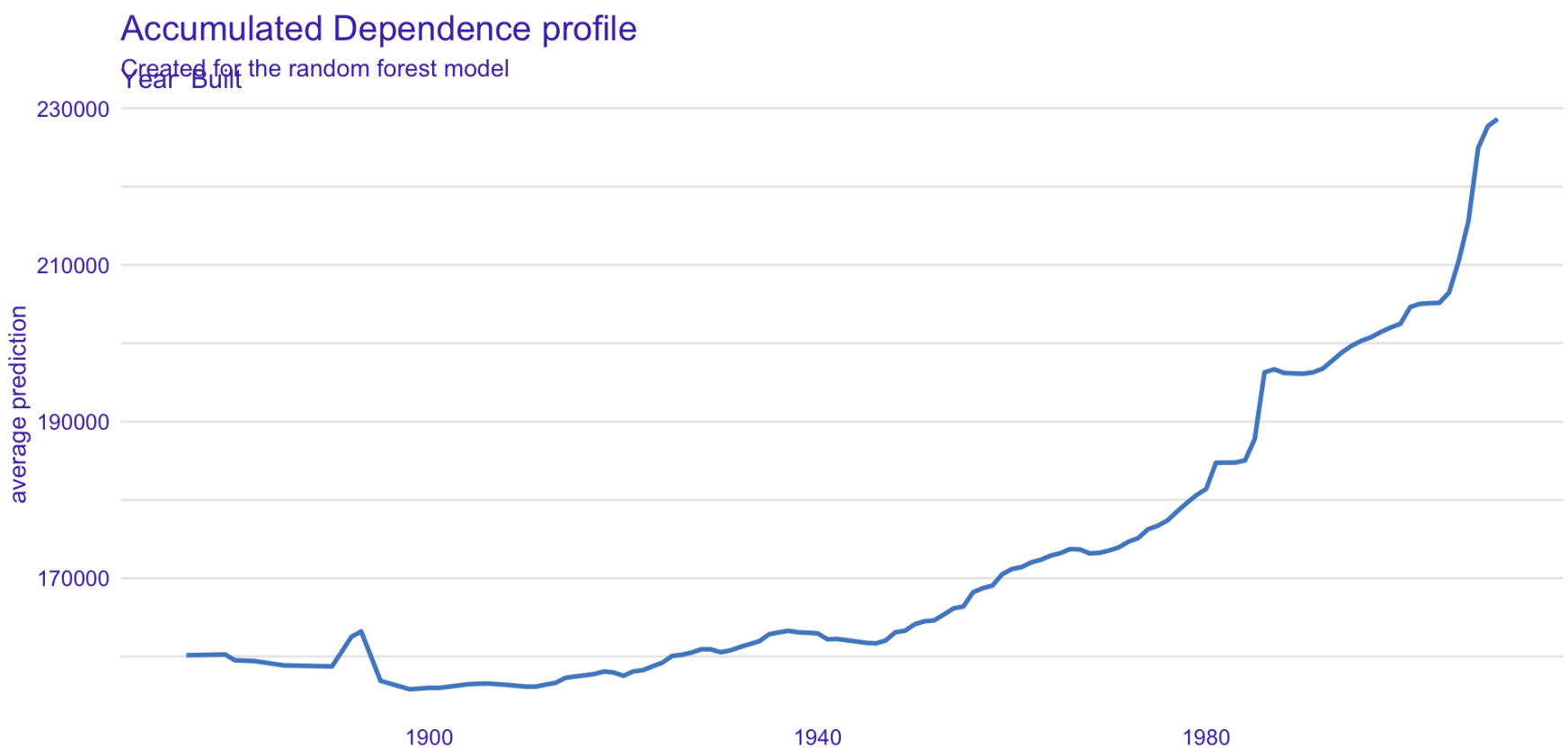
1. Puede solamente mostrar una sola variable con sentido.
2. ICE tiene el mismo problema que PDP si la variable de interés está correlacionada con las otras algunos puntos en las lineas pueden ser puntos sin sentido.
3. Si hay muchas curvas puede ser muy confuso, se puede usar transparencias o dibujar una muestra de lineas

Efecto local acumulado (ALE)

- ALE describe como las variables explicativas influyen la predicción del ML en promedio.
- Los gráficos ALE son rápidos y una alternativa insesgada a PDP.
- Tiene el mismo objetivo que el PDP pero trata de resolver una de las debilidades del PDP que es cuando x_C y x_S están correlacionadas.

Ejemplo: Datos Ames

```
1  ale_age <- model_profile(explainer_rf, N = 500,
2                           variables = "Year_Built",
3                           type="accumulated")
4  plot(ale_age)
```



Ejemplo: Datos Ames

Alternativamente explorar el paquete ALEPlot que permite hacer el ale y además el pdp

Ejemplo: Datos Ames

Dentro de `tidymodels` lo que usamos para interpretabilidad es el paqueta `DALEX` y `DALEXtra`

Página del paquete https://dalex.drwhy.ai

Página del libro https://ema.drwhy.ai

Tu turno

En alguno de los modelos de ejemplo que tenés en tu proyecto comienza a analizar la interpretabilidad con las herramientas vistas