Добавляем предикторы

Как обойтись без моделей?

Обойтись без моделей: план

- Как переделать временные ряды в перекрестные данные?
- Добавить лаги переменной y_t.
- Использовать агрегирующие функции и скользящее или растущее окно.

Как обойтись без моделей?

Старые друзья

Есть алгоритмы, которые по обучающей выборке зависимой переменной y, обучающей матрице предикторов X, и новым предикторам X_F строят прогноз $\hat{y}_F$.

Случайный лес, градиентный бустинг… и даже обычная регрессия!

Можно усреднять прогнозы ARIMA/ETS и прогнозы других алгоритмов.

Как создать предикторы?

Из одного столбца y можно создать целую матрицу X предикторов!

- Использовать лаги y_{t-k}.
- Использовать функции от лагов в качестве предикторов.

Используем лаги y

Для примера возьмём два лага, Ly_t и L^2y_t.

Обучающая выборка:

$$\begin{pmatrix} y_3 \\ y_4 \\ y_5 \\ \vdots \\ y_T \end{pmatrix} \quad \begin{pmatrix} y_1 & y_2 \\ y_2 & y_3 \\ y_3 & y_4 \\ \vdots & \vdots \\ y_{T-2} & y_{T-1} \end{pmatrix}$$

Выборка для прогнозирования:

$$\begin{pmatrix} ? \end{pmatrix} \quad \begin{pmatrix} y_{T-1} & y_T \end{pmatrix}$$

Сколько лагов добавить?

- Каждый добавленный лаг сокращает обучающую выборку!
- Разумно добавить ближайшие лаги Ly_t, L^2y_t.
- Для сезонных данных разумно добавить сезонный лаг $L^{12}y_t$.
- Есть алгоритмы чувствительные к лишним предикторам: например, регрессия.
- Есть алгоритмы нечувствительные к лишним предикторам: например, случайный лес.

Функции от лагов

При прогнозировании y_t честно использовать любую функцию от предыдущих $y_{t-1}, y_{t-2}, \ldots$

Например:

- $\Delta y_{t-1} = y_{t-1} - y_{t-2}$;
- $\max\{y_{t-1}, y_{t-2}, y_{t-3}\}$;
- $\min\{y_{t-1}, y_{t-2}, \ldots, y_1\}$.

Типичный предиктор

- Агрегирующая функция:
  Минимум, максимум, среднее, медиана, размах, выборочная дисперсия, выборочное стандартное отклонение, $\ldots$
- Аргумент агрегирующей функции:
  Скользящее окно: агрегирующая функция применяется, скажем, к трём предыдущим значениям y_{t-1}, y_{t-2}, y_{t-3}.
  Растущее окно: агрегирующая функция применяется ко всем предыдущим значениям y_{t-1}, y_{t-2}, $\ldots$, y_1.

Используем функции лагов y

Для примера возьмём максимум скользящим окном и минимум растущим окном.

Обучающая выборка:

$$\begin{pmatrix} y_3 \\ y_4 \\ y_5 \\ \vdots \\ y_T \end{pmatrix} \quad \begin{pmatrix} \max\{y_1, y_2\} & \min\{y_1, y_2\} \\ \max\{y_2, y_3\} & \min\{y_1, y_2, y_3\} \\ \max\{y_3, y_4\} & \min\{y_1, \ldots, y_4\} \\ \vdots & \vdots \\ \max\{y_{T-2}, y_{T-1}\} & \min\{y_1, \ldots, y_{T-1}\} \end{pmatrix}$$

Выборка для прогнозирования:

$$\begin{pmatrix} ? \end{pmatrix} \quad \begin{pmatrix} \max\{y_{T-1}, y_T\} & \min\{y_1, \ldots, y_T\} \end{pmatrix}$$

Обойтись без моделей: итоги

- Помните о случайном лесе, градиентном бустинге и даже об обычной регрессии.
- Добавьте лаги зависимой переменной.
- Добавьте агрегирующие функции скользящим и растущим окном.

У нас есть ещё время!

У нас ещё есть время: план

- Предикторы тренда.
- Сезонные и праздничные дамми.
- Косинусы и синусы.

Используем время!

Для примера возьмём t и $\sqrt{t}$.

Обучающая выборка:

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_T \end{pmatrix} \qquad \begin{pmatrix} 1 & \sqrt{1} \\ 2 & \sqrt{2} \\ 3 & \sqrt{3} \\ \vdots & \vdots \\ T & \sqrt{T} \end{pmatrix}$$

Выборка для прогнозирования:

$$\begin{pmatrix} ? \end{pmatrix} \qquad \begin{pmatrix} T+1 & \sqrt{T+1} \end{pmatrix}$$

Включать ли монотонные преобразования времени?

- Всегда можно попробовать включить!
- Алгоритмам основанным на построении деревьев (случайные лес, градиентный бустинг) дополнительные монотонные преобразования времени бесполезны.
- Помните о возможном преобразовании исходной переменной (логарифм, преобразование Бокса-Кокса).

Сезонные и праздничные дамми

Если сезонов немного, то разумно включить дамми на каждый сезон.

Обучающая выборка для квартальных данных:

$$
\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \\ y_6 \\ \vdots \\ y_T \end{pmatrix}
\qquad
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 1 & 0 \end{pmatrix}
$$

Ловушка дамми-переменных

В регрессии помните о ловушке дамми-переменных!

- Либо дамми на каждый сезон и модель без константы.
- Либо дамми на все сезоны кроме одного и модель с константой.

Алгоритмы основанные на построении деревьев (случайные лес, градиентный бустинг) устойчивы к ловушке дамми.

Зачем нужны синусы и косинусы?

Стратегия добавления всех дамми переменных плохо работает, если их нужно много.

Вряд ли стоит добавлять 365 дамми-переменных для дневных данных.

Обойтись малым числом предикторов помогут синус и косинус!

Два факта:

- Период у $\sin t$ и $\cos t$ равен 2π;
- При умножении аргумента на a период сокращается в a раз.

Разложение Фурье

Теорема

Любая непрерывная и дифференциируемая функция f с периодом 2π может быть представлена в виде

$$f(t) = c + \sum_{k=1}^{\infty} a_k \cos(kt) + b_k \sin(kt).$$

Практический рецепт для дневных данных:

- Добавьте предикторы $\cos\left(\frac{2\pi}{365} \cdot t\right)$ и $\sin\left(\frac{2\pi}{365} \cdot t\right)$;
- Добавьте предикторы $\cos\left(\frac{2\pi}{365} \cdot 2t\right)$ и $\sin\left(\frac{2\pi}{365} \cdot 2t\right)$;
- Добавьте предикторы $\cos\left(\frac{2\pi}{365} \cdot 3t\right)$ и $\sin\left(\frac{2\pi}{365} \cdot 3t\right)$;
- $\ldots$

У нас есть ещё время: итоги

- Используйте время в качестве предиктора.
- Сезонность в предикторах можно отразить с помощью дамми-переменных или с помощью косинуса и синуса.

Случайный лес и градиентный бустинг

Случайный лес и градиентный бустинг: план

- Деревья для случайного леса.
- Деревья для градиентного бустинга.

Случайный лес

Особенности деревьев:

- Деревьев очень много, $n_{tree} = 10000$.
- Деревья очень ветвистые.
- Деревья равноправные: лес усредняет прогнозы деревьев.

Почему деревья разные?

Два источника случайности:

- Каждое дерево растёт случайно.
  Каждое дерево в каждом своём узле из большого исходного списка предикторов удаляет случайным образом часть предикторов перед поиском оптимального деления.
- Каждое дерево растёт на своей бутстрэп выборке.
  Из исходных T наблюдений перед построением дерева создаётся новая искусственная выборка с возвращением в T наблюдений.

Бутстрэп выборка

Исходная выборка:

$$y = \begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \\ y_5 \end{pmatrix} \qquad X = \begin{pmatrix} a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_3 \\ a_4 & b_4 \\ a_5 & b_5 \end{pmatrix}$$

Пример бутстрэп выборки:

$$y^* = \begin{pmatrix} y_4 \\ y_2 \\ y_1 \\ y_4 \\ y_1 \end{pmatrix} \qquad X^* = \begin{pmatrix} a_4 & b_4 \\ a_2 & b_2 \\ a_1 & b_1 \\ a_4 & b_4 \\ a_1 & b_1 \end{pmatrix}$$

Лес в формулах

Построение деревьев:

- Из (y, X) случайно создаём первую бутстрэп выборку (y^*, X^*).
  Первое дерево T_1 учится прогнозировать y^* с помощью X^*.
- Из (y, X) случайно создаём вторую бутстрэп выборку (y^*, X^*).
  Второе дерево T_2 учится прогнозировать y^* с помощью X^*.
- $\ldots$

Построение прогнозов:

$$\text{Forest}(x) = \frac{1}{n_{tree}}(T_1(x) + T_2(x) + \ldots + T_{n_{tree}}(x))$$

Градиентный бустинг

Особенности деревьев:

- Деревьев относительно мало, вполне возможно, $n_{tree} = 1000$.
- Деревья не очень ветвистые.
- Каждое последующее дерево уточняет суммарный прогноз предыдущих деревьев с весом η.

Бустинг в формулах

Построение деревьев и прогнозов:

- Без деревьев прогноз тривиален:
$$GB_0(x) = \bar{y} = \frac{y_1 + y_2 + \ldots + y_T}{T}$$
- Первое дерево T_1 учится прогнозировать $r_0 = y - GB_0(X)$ с помощью X.
- Составляем композитный прогноз:
$$GB_1(x) = GB_0(x) + \eta T_1(x) = \bar{y} + \eta T_1(x).$$
- Второе дерево T_2 учится прогнозировать $r_1 = y - GB_1(X)$ с помощью X.
- Составляем композитный прогноз:
$$GB_2(x) = GB_1(x) + \eta T_2(x) = \bar{y} + \eta(T_1(x) + T_2(x)).$$
- …

Как сравнить алгоритмы?

- Нельзя применять критерий Акаике AIC.
- Можно применять кросс-валидацию!

  Можно даже применять кросс-валидацию для перекрестных данных!

Случайный лес и градиентный бустинг: итоги

- У алгоритмов куча вариаций.
- В лесу много равноправных деревьев и они ветвистые.
- В градиентном бустинге деревьев мало, они небольшие и каждое последующее дерево уточняет прогнозы.
- Для случайного леса стоит взять побольше деревьев.
- В градиентном бустинге важен баланс числа деревьев и темпа обучения.

Предикторы и $ARIMA$

Предикторы и $ARIMA$: план

- Регрессия с $ARMA$ ошибками.
- $ARMAX$ модель.
- $ARDL$ модель.

Регрессия с $ARMA$ ошибками

Уравнение

$$y_t = \beta_1 + \beta_2 a_t + \beta_3 b_t + \varepsilon_t,$$

где a_t и b_t — предикторы.

- Ряды (y_t), (a_t), (b_t), (ε_t) стационарны.
- $\varepsilon_t \sim ARMA(p, q)$ относительно белого шума (u_t).
- $\mathbb{E}(\varepsilon_t \mid a_{t-1}, b_{t-1}, a_{t-2}, b_{t-2}, \ldots) = 0$.
- Четвертые моменты предикторов конечны.

$ARMAX$ модель

Уравнение

$$y_t = c+\gamma_1 a_t+\gamma_2 b_t+\beta_1 y_{t-1}+\ldots+\beta_p y_{t-p}+u_t+\alpha_1 u_{t-1}+\ldots+\alpha_q u_{t-q},$$

где a_t и b_t — предикторы, а (u_t) — белый шум.

- Ряды (y_t), (a_t), (b_t) стационарны.
- $\mathbb{E}(u_t \mid a_{t-1}, b_{t-1}, y_{t-1}, a_{t-2}, b_{t-2}, y_{t-2}, \ldots) = 0$.
- Четвертые моменты предикторов конечны.

$ARMAX$ модель не полностью эквивалентна регрессии с $ARMA$ ошибками, но качество прогнозов у моделей примерно одинаковое.

Свойства $ARMAX$ модели и регрессии с $ARMA$ ошибками

1. Если выполнены предпосылки, то оценки метода максимального правдоподобия состоятельны.
2. Для нестационарных переменных (y_t) и предикторов (a_t) и (b_t) можно перейти к первым разностям.
3. Оценки останутся состоятельны, если добавить тренд, дамми на сезонность и тригонометрические предикторы.
4. Не любой предиктор даёт возможность получить состоятельную оценку коэффициента.
5. Иногда можно получить хорошие прогнозы, даже если предпосылки нарушены.

$ARDL$ модель

ARDL — AutoRegressive Distributed Lag model

Авторегрессионная модель с распределёнными лагами.

Уравнение $ARDL(p,q)$ модели

$$y_t = c + \beta_1 y_{t-1} + \ldots + \beta_p y_{t-p} + x_t + \alpha_1 x_{t-1} + \ldots + \alpha_q x_{t-q} + u_t$$

- Ошибки (u_t) являются белым шумом.
- Процесс (x_t) или процесс (Δx_t) является стационарным.
- Процесс (y_t) является нестационарным, но (Δy_t) является стационарным.
- $\mathbb{E}(u_t \mid y_{t-1}, x_{t-1}, y_{t-2}, x_{t-2}, \ldots) = 0$.

Свойства $ARDL$ модели

- Вместо лагов шума (u_t) лаги предиктора (x_t).
- Подходит для нестационарного (y_t).
- Используют для нахождения долгосрочных взаимоотношений между рядами.
- Если предпосылки выполнены, то оценки МНК состоятельны, хотя и смещены.
- Можно добавить несколько предикторов с разным числом лагов.

Предикторы и $ARIMA$: итоги

- Для стационарных данных можно использовать регрессию с $ARMA$ ошибками или $ARMAX$ модель.
- Регрессию с $ARMA$ ошибками можно строить в разностях.
- Для нестационарных рядов иногда можно использовать $ARDL$ модель.