Интерпретируй это. Как показывать ML-модели

Дмитрий Колодезев
ООО Промсофт, Новосибирск

Data Fest Siberia 28.09.2019

Уведомление

НАСТОЯЩАЯ ПРЕЗЕНТАЦИЯ НЕ РЕКОМЕНДУЕТСЯ ДЛЯ ПРОСМОТРА ДЕТЯМ И ПОДРОСТКАМ ДО 18 ЛЕТ, БЕРЕМЕННЫМ И КОРМЯЩИМ ЖЕНЩИНАМ, ЛИЦАМ С ЗАБОЛЕВАНИЯМИ ЦЕНТРАЛЬНОЙ НЕРВНОЙ СИСТЕМЫ, ПОЧЕК, ПЕЧЕНИ И ДРУГИХ ОРГАНОВ ПИЩЕВАРЕНИЯ.

Проблема

- Молодой динамичный стартап
- Приближается корпоратив
- Нужно выбрать вино
- Раньше выбирал технический директор
- Нужно заменить его на ML-модель
- https://archive.ics.uci.edu/ml/datasets/Wine+Quality

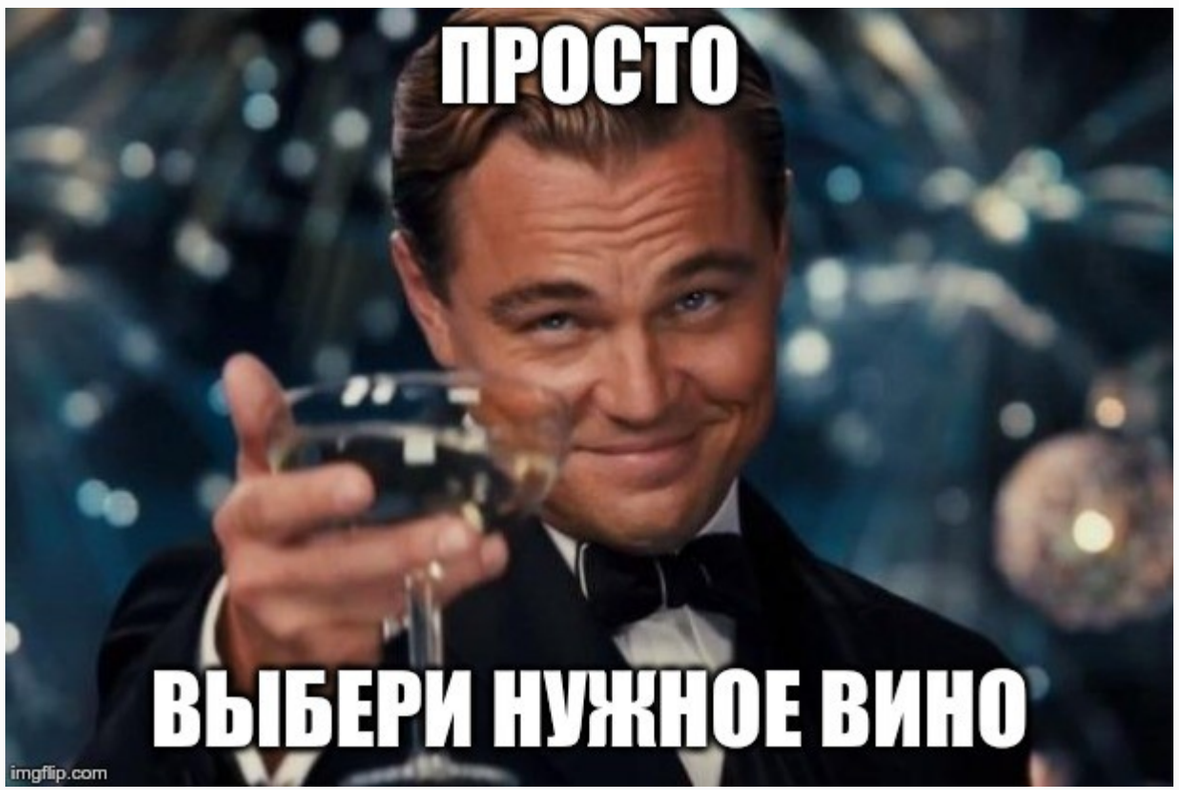
Берем и решаем

fixed acidity	volatile acidity	citric acid	residual sugar	chlorides	free sulfur dioxide	total sulfur dioxide	density	pH	sulphates	alcohol	quality	wine_type
7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	red
7.8	0.88	0.00	2.6	0.098	25.0	67.0	0.9968	3.20	0.68	9.8	5	red
7.8	0.76	0.04	2.3	0.092	15.0	54.0	0.9970	3.26	0.65	9.8	5	red
11.2	0.28	0.56	1.9	0.075	17.0	60.0	0.9980	3.16	0.58	9.8	6	red
7.4	0.70	0.00	1.9	0.076	11.0	34.0	0.9978	3.51	0.56	9.4	5	red

```
model = CatBoostRegressor(iterations=500)
model.fit(X_train, y_train, cat_features=[11])
h = model.predict(X_test)
mean_squared_error(y.iloc[test_index], h)
```

0.4819942217614274

Задача практически решена

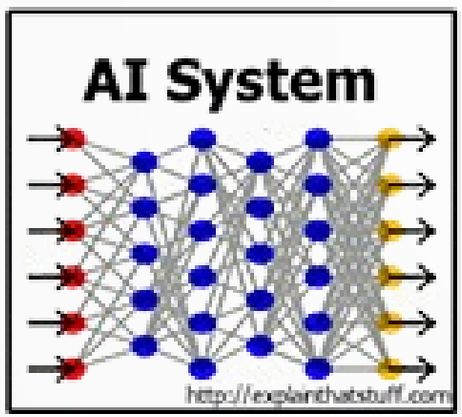


Нет доверия — нет внедрения

Принимаем модель на работу

- Что она умеет?
- Что она не умеет?
- Как она принимает решения?
- Где она накосячит?
- Пользователь строит модель модели
- И эту модель тоже нужно обучать

DARPA, eXplainable AI



- We are entering a new age of AI applications
- Machine learning is the core technology
- Machine learning models are opaque, non-intuitive, and difficult for people to understand

DoD and non-DoD Applications

Transportation

Security

Medicine

Finance

Legal

Military

- Why did you do that?
- Why not something else?
- When do you succeed?
- When do you fail?
- When can I trust you?
- How do I correct an error?

Explainable Artificial Intelligence (XAI)

Как объяснить?

- Использовать простую модель
- Показать на примерах
- Показать внутренности
- Построить суррогатную модель
  - Локальную — для одной точки
  - Глобальную — для всего набора данных
- pip install shap (lime, anchor_exp, alibi...)

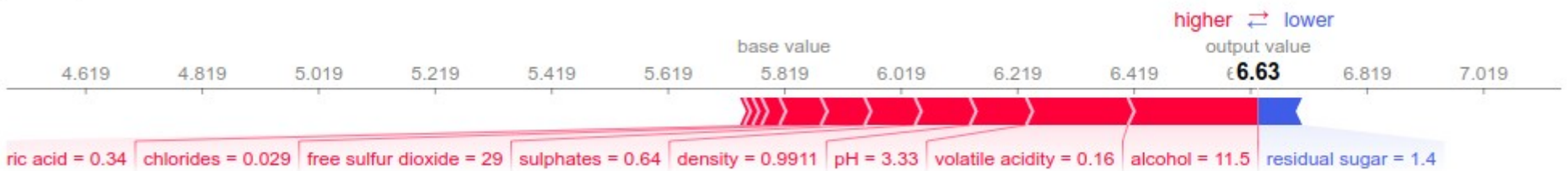
Shapley values

- Как делить мамонта между охотниками?
- Как делить позор проигрыша на Kaggle?
- Кооперативные игры.

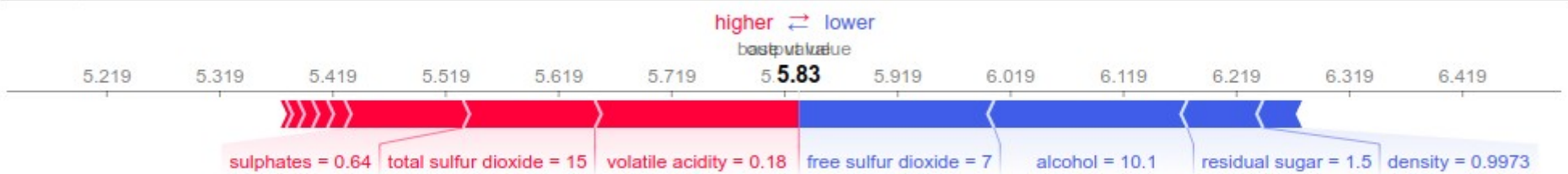
```
shap_values = model.get_feature_importance(
    Pool(X.iloc[test_index],
         label=y.iloc[test_index],cat_features=cat_features), type="ShapValues")
expected_values = shap_values[:, -1]
shap_values = shap_values[:, :-1]
scaler = preprocessing.RobustScaler()
scaled_shap_values = scaler.fit_transform(shap_values)
shap.initjs()
```

Алкоголь — это хорошо?

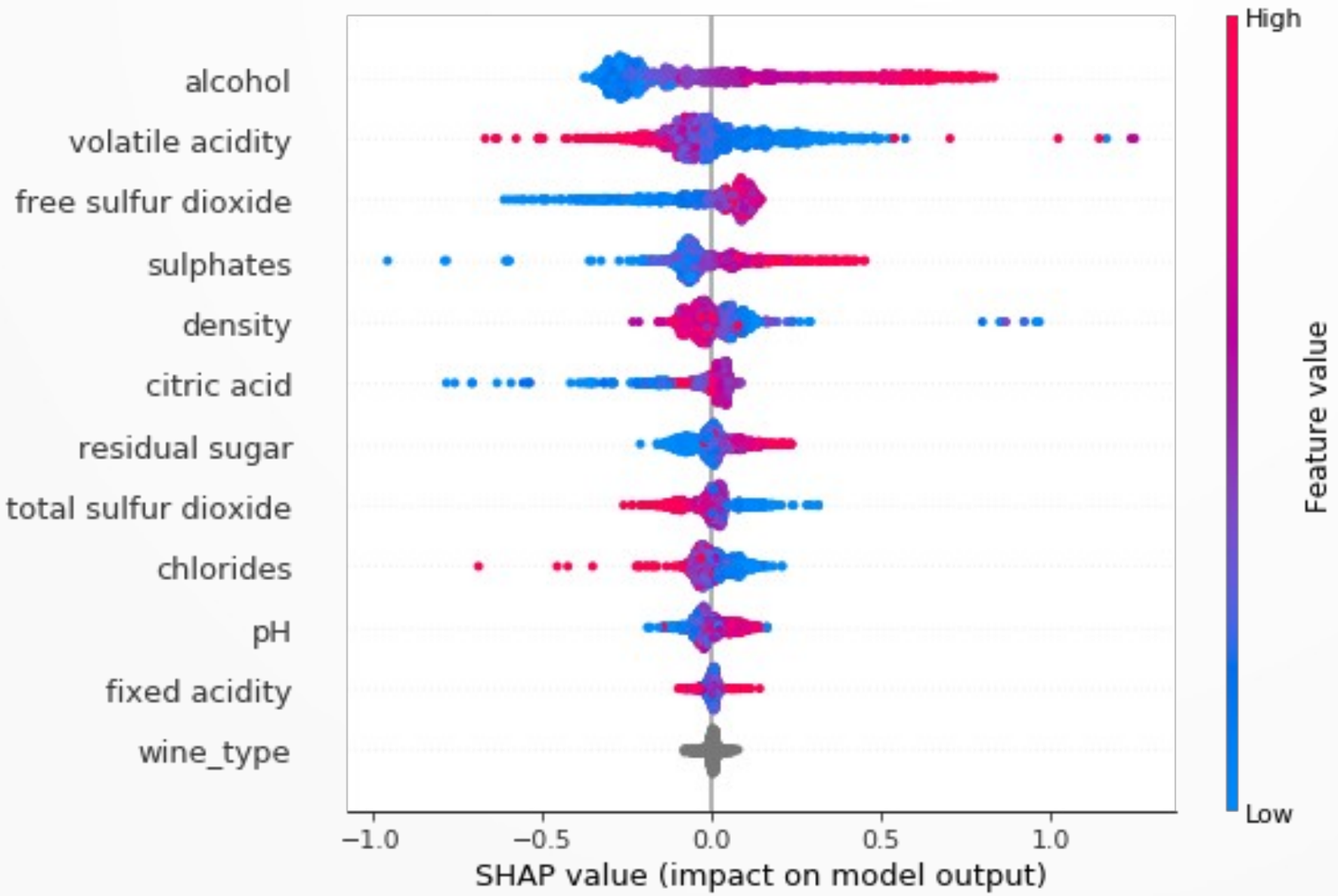
```
1  shap.force_plot(expected_values[0], shap_values[0,:], X.iloc[test_index[0], :])
```




```
1  shap.force_plot(expected_values[0], shap_values[1,:], X.iloc[test_index[1], :])
```



Сразу не скажешь



Алкоголь — это плохо!

- Люди оценивают вкус, запах, кислинку, «округлость», цвет, крепость, цену, надпись на этикете.
- Непонятные заказчику признаки
- Объяснения модели для разных точек противоречат друг другу
- Попытка не засчитана.

Что у нас в данных

- pH — кисленькое
- volatile acidity — ацетоном пахнет
- alcohol — содержание спирта
- residual sugar — сахар
- wine_type — красное или белое

Заказчик на вкус не различит:

- fixed acidity, citric acid, chlorides, sulphates, free sulfur dioxide, total sulfur dioxide, density

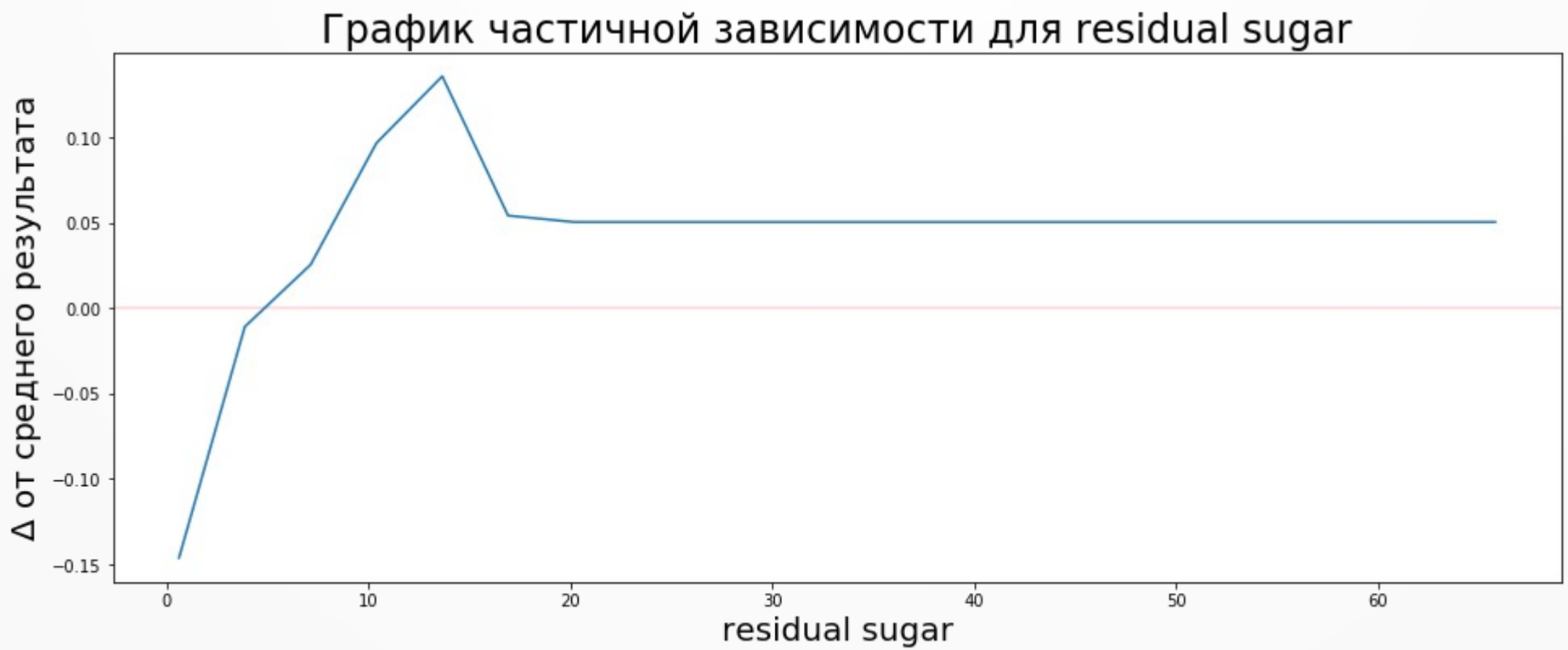
Partial Dependence Plot

```
def calc_p(model, X, col, v):
    tmp = X.copy()
    tmp[col] = v
    h = model.predict(tmp)
    return np.mean(h)

def pdp_numeric(X, col, model):
    tx = np.linspace(X[col].min(), X[col].max(), 21)
    m = np.mean(model.predict(X))
    ty = [calc_p(model, X, col, t) - m for t in tx]
    plt.figure(figsize=(16, 6))
    plt.axhline(0, linewidth=0.25, color='r')
    plt.title("График частичной зависимости для {}".format(col), fontsize=24)
    plt.xlabel(col, fontsize=20)
    plt.ylabel('$\Delta$ от среднего результата', fontsize=20)
    plt.plot(tx, ty);
```

Partial Dependence Plot - 1

pdp_numeric(X.iloc[test_index], 'residual sugar', model)

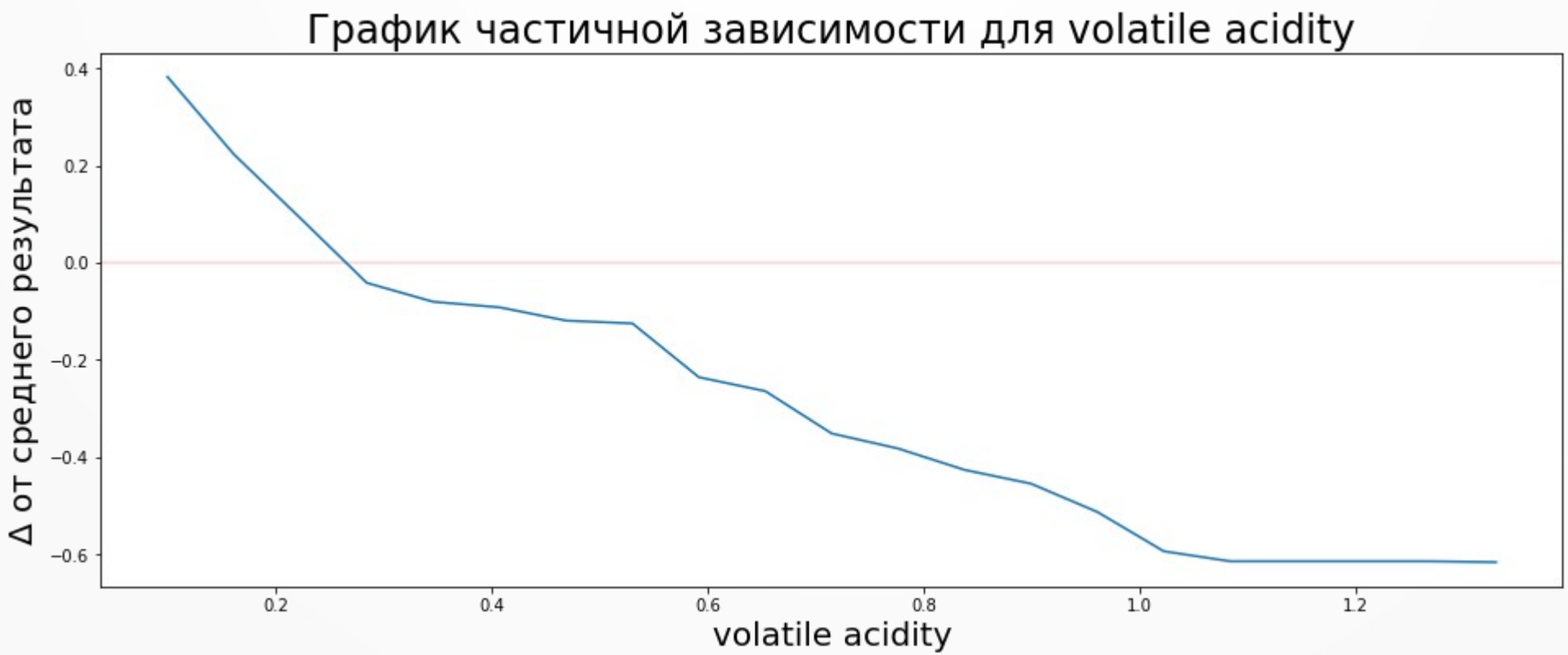


Partial Dependence Plot - 2

pdp_numeric(X.iloc[test_index], 'alcohol', model)

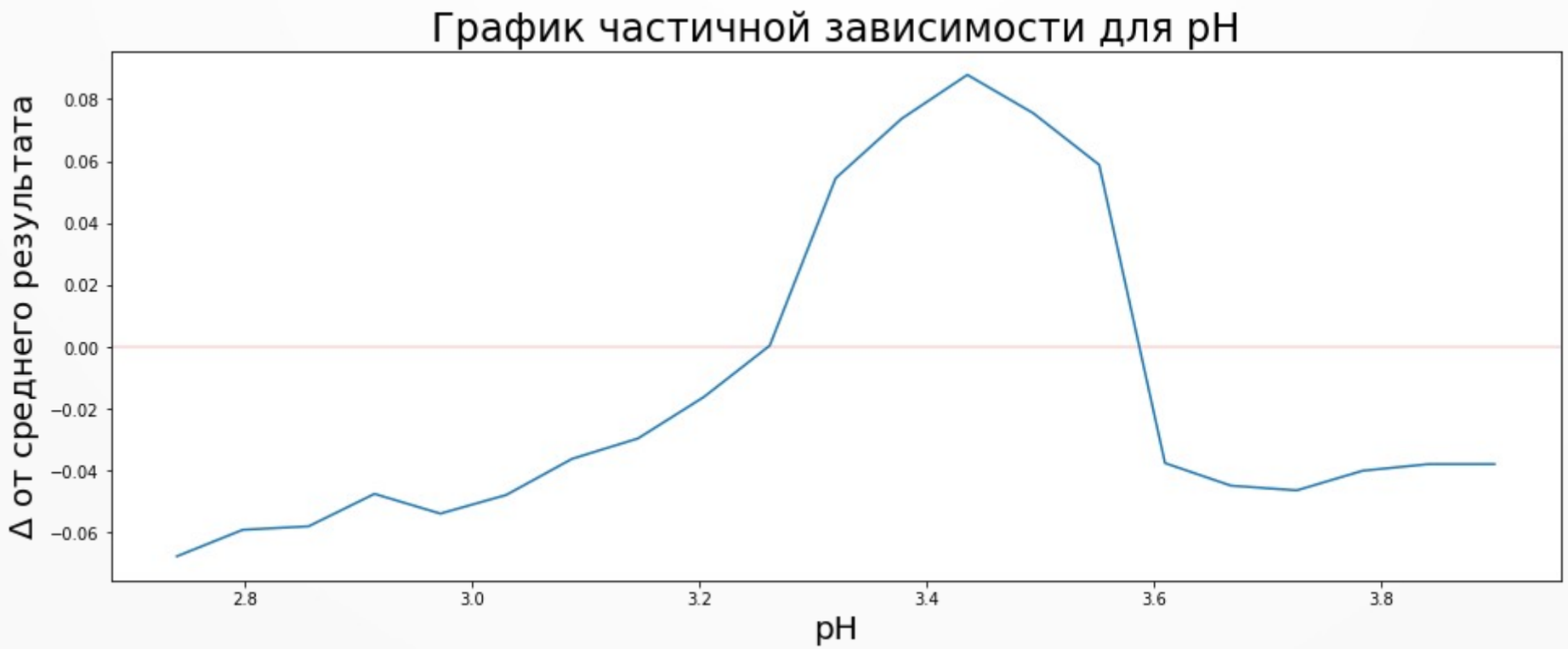
Partial Dependence Plot - 3

pdp_numeric(X.iloc[test_index], 'volatile acidity', model)



Partial Dependence Plot - 4

pdp_numeric(X.iloc[test_index], 'pH', model)



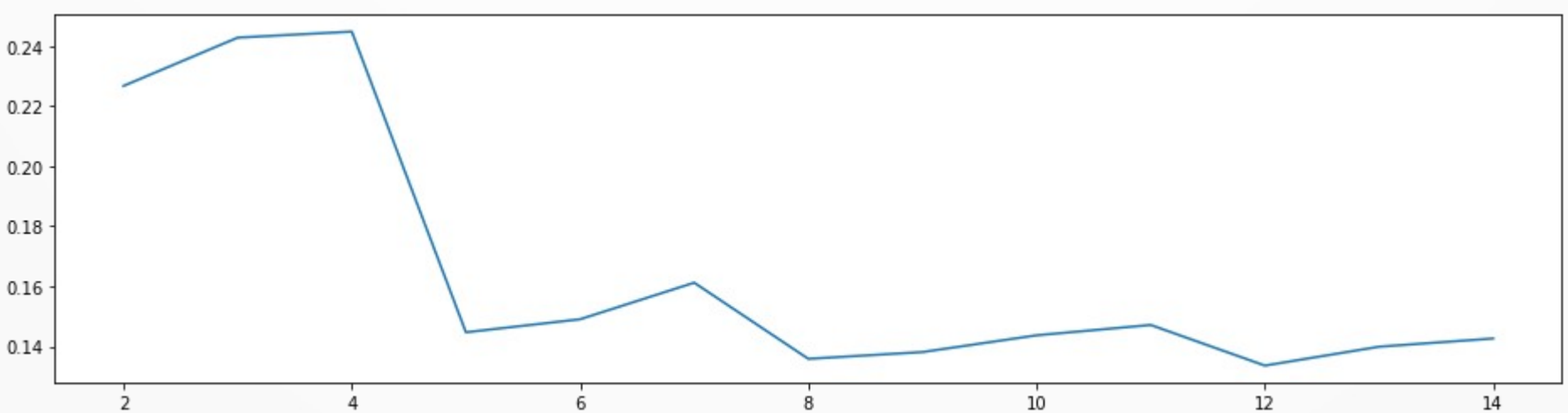
Группируем объяснения

- Мы можем собрать кластеры с «похожими» объяснениями
- Тогда внутри одного кластера объяснения будут похожими.
- Найти, сколько кластеров нам нужно
- Найти в каждом кластере точки, знакомые пользователю
- Построить модель — в какой кластер попадет

Пяти кластеров хватит всем

```
from sklearn.cluster import AgglomerativeClustering
from sklearn.metrics import silhouette_score

score = []
n_clusters = [i for i in range(2, 15)]
clustering_params = dict(affinity='euclidean', linkage='ward')
for n in n_clusters:
    cluster = AgglomerativeClustering(n_clusters=n, **clustering_params)
    shap_clusters = cluster.fit_predict(scaled_shap_values)
    score.append(silhouette_score(scaled_shap_values, shap_clusters))
```



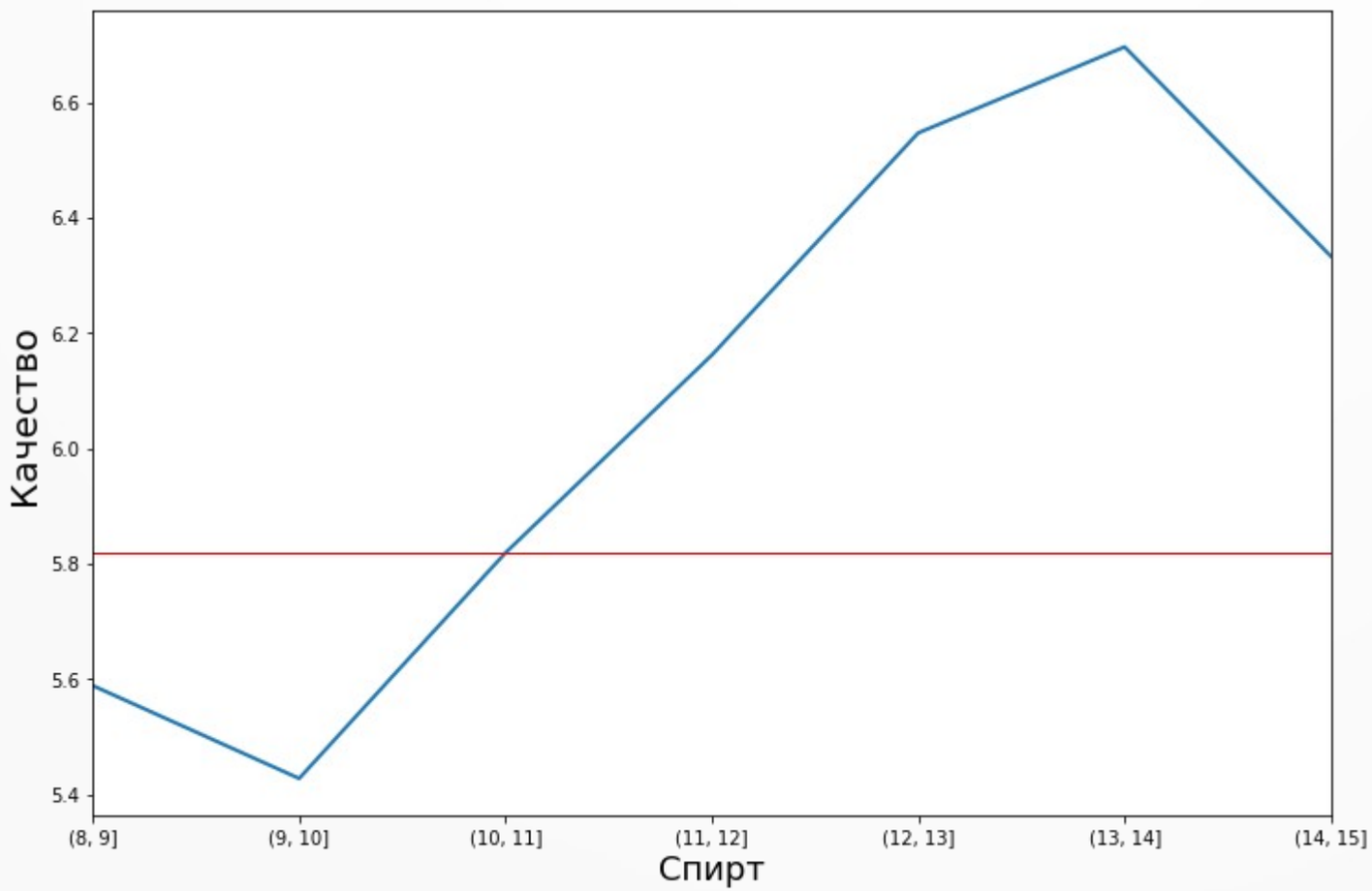
Mean target plot

- Модель просто воспроизводит данные
- Давайте напрямую посмотрим в данные
- График усредненной зависимости целевого признака («таргет на бинах»)
- Покажет нам, какие зависимости в данных мы скорее всего выучим

```
tmp = df.copy()
tmp['BINS'] = pd.cut(tmp['alcohol'], bins=[x for x in np.arange(8, 16)])
fig = plt.figure(figsize=(12, 8))
tmp.groupby('BINS').quality.mean().plot(linewidth=2);
plt.ylabel('Качество', fontsize=20)
plt.xlabel('Спирт', fontsize=18);
plt.axhline(df.quality.mean(), linewidth=1, color='r');
```

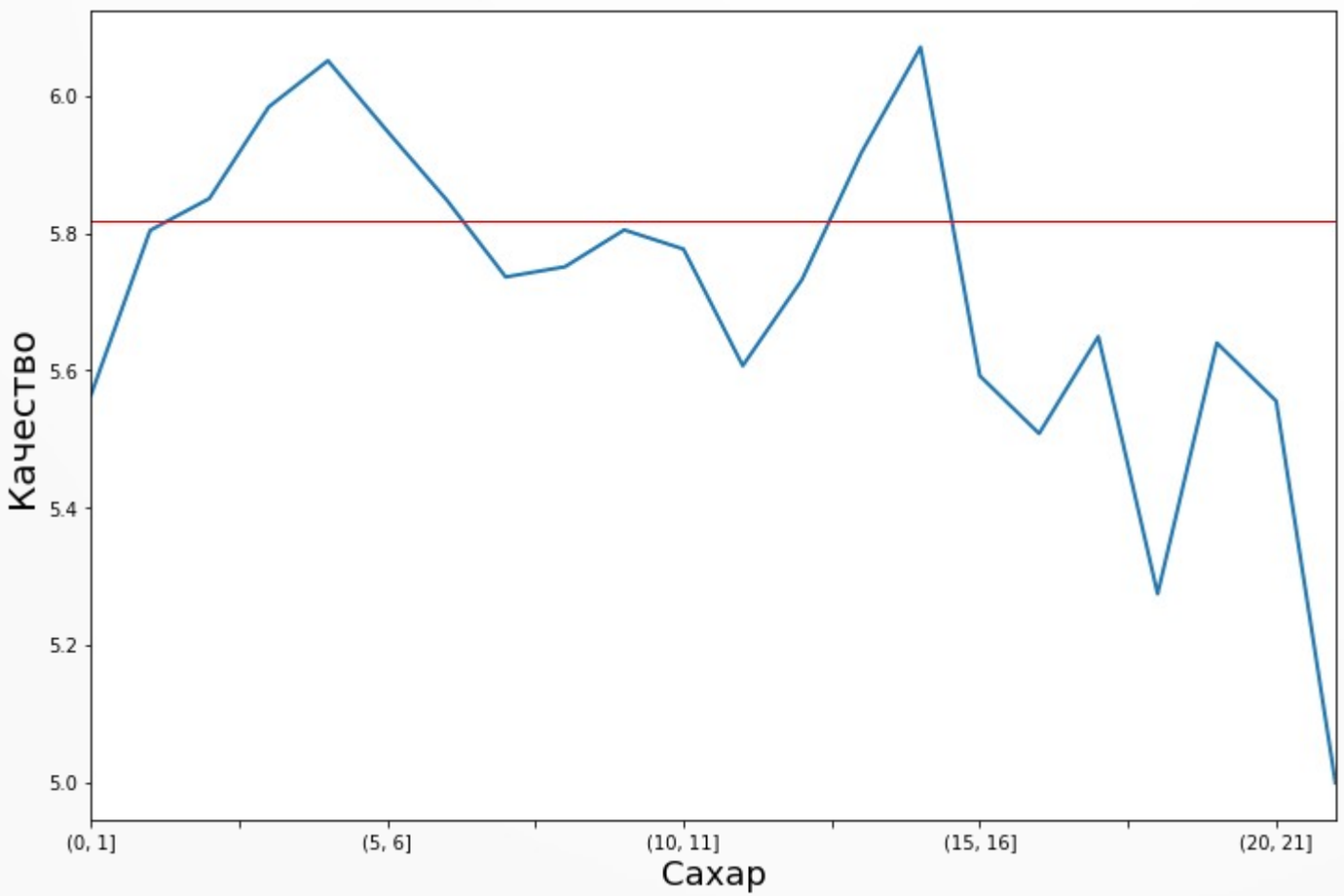
Mean target plot - 1

```
tmp['BINS'] = pd.cut(tmp['alcohol'], bins=[x for x in np.arange(8, 16)])
```



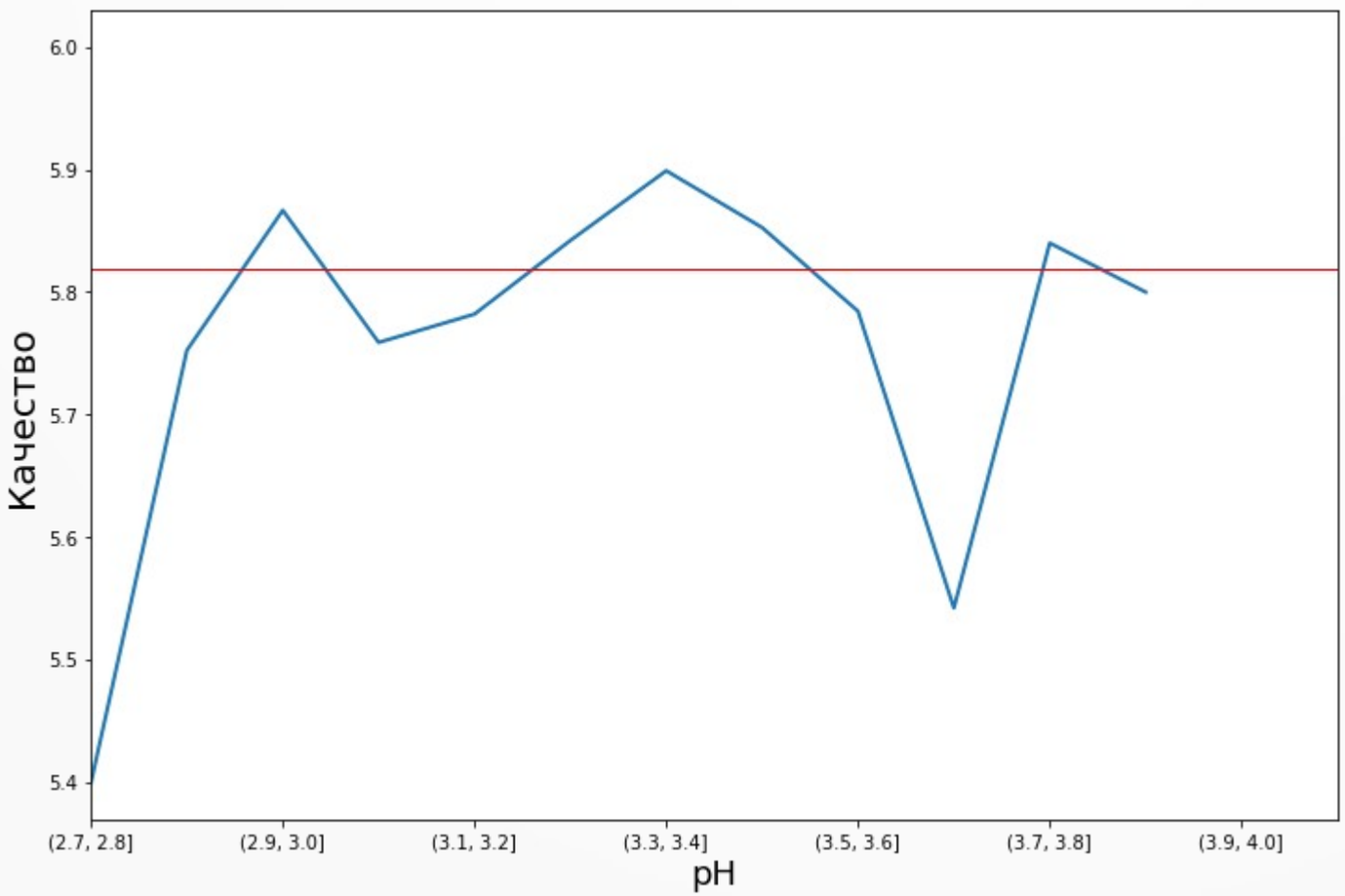
Mean target plot - 2

```
tmp['BINS'] = pd.cut(tmp['residual sugar'], bins=[x for x in np.arange(0, 23)])
```



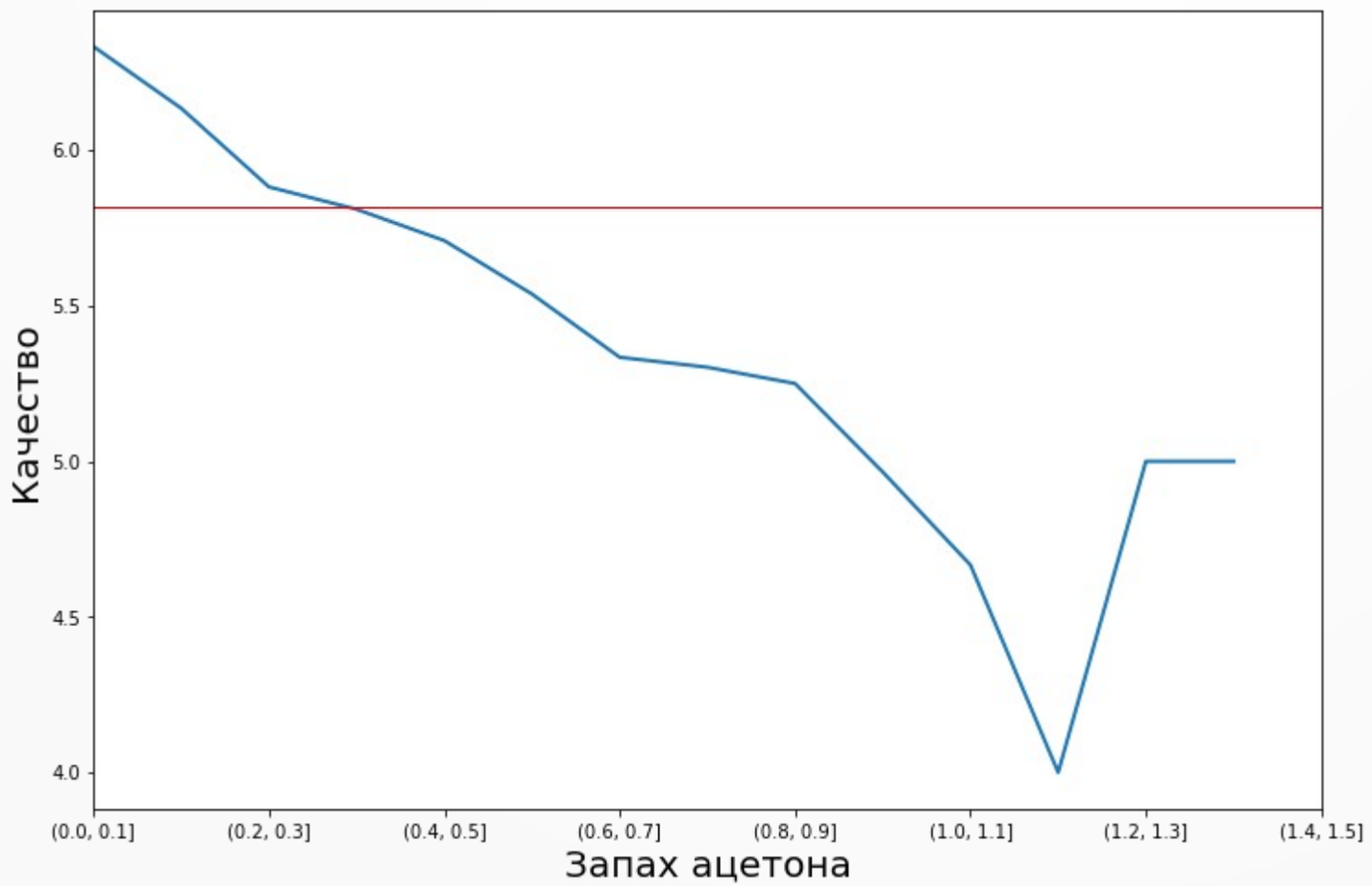
Mean target plot - 3

```
tmp['BINS'] = pd.cut(tmp['pH'], bins=[x for x in np.arange(2.7, 4.2, 0.1)])
```



Mean target plot - 4

```
tmp['BINS'] = pd.cut(tmp['volatile acidity'], bins=[x for x in np.arange(0, 1.6, 0.1)])
```



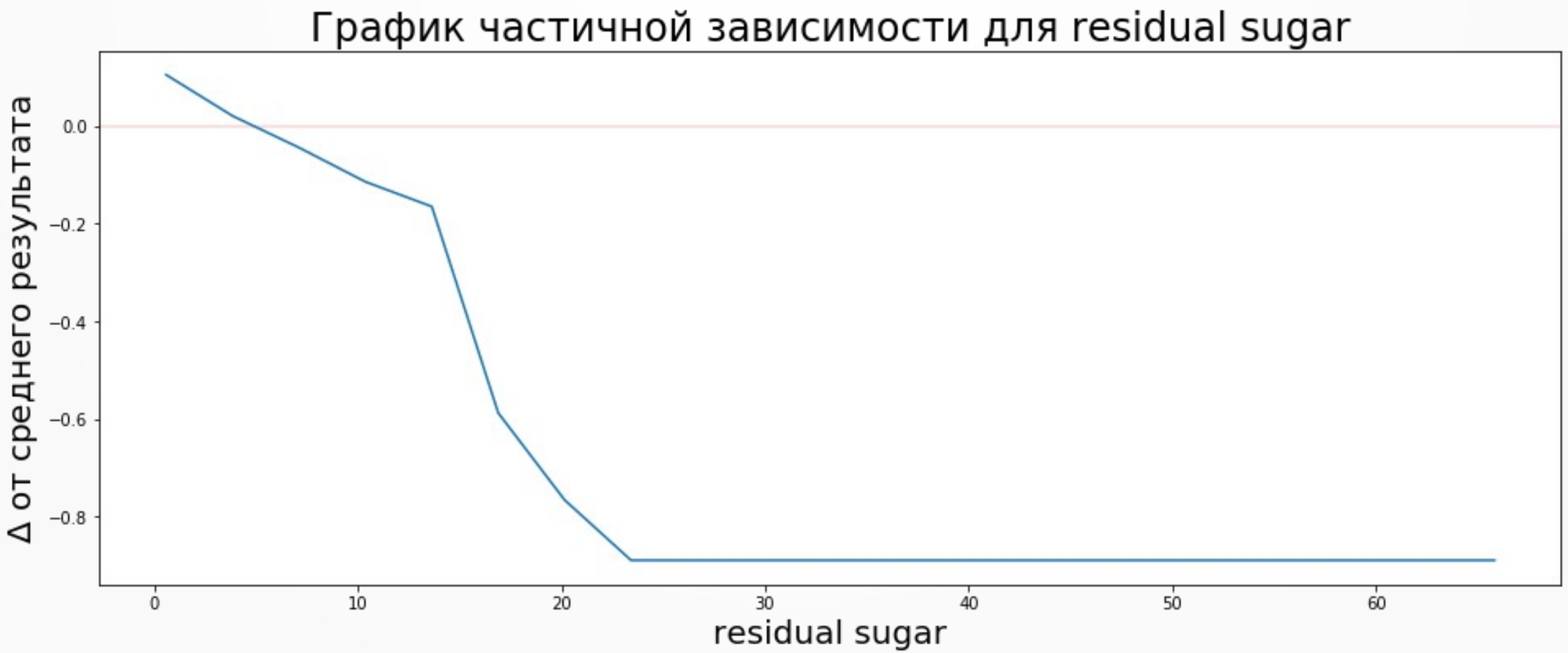
Все-таки, хорошо или нет?

Требуйте монотонности

Добавляем убеждения заказчика в модель (множественность хороших моделей)

```
monotonicity = {
    'residual sugar': -1,
    'alcohol' : 1,
    'volatile acidity': -1
}
model = CatBoostRegressor(
    iterations=5000,
    random_seed=20190927,
    monotone_constraints=[monotonicity.get(c, 0) for c in X.columns],
    model_shrink_rate=0, # https://github.com/catboost/catboost/issues/994
    task_type="CPU")
```

Монотонный сахар



Итог

- Объяснили модель с помощью Shapley Values
- Выделили похоже объясняемые кластеры
- Показали, как признак влияет на результат
- Показали, как таргет зависит от данных
- Построили более понятную монотонную модель.

Почитать

- Дьяконов, Интерпретации чёрных-ящиков
- Becker, Machine Learning Explainability
- Molnar, Interpretable Machine Learning
- CVPR 2018 Tutorial
- ICCV 2019 Tutorial
- MIT Network Dissection
- 1-я часть доклада
- 2-я часть доклада

Вопросы?

Слайды тут

dkolodezev

promsoft

dkolodezev

d_key

dmitry_kolodezev

https://kolodezev.ru/download/slides-interpretation-v3.pdf