Задание 6. Логистическая регрессия
(подробней см. в [1, стр. 93–113], [2, стр. 187–197], [3, стр. 62–67], [4, 5])
(документация: API Reference — scikit-learn 1.1.3 documentation
sklearn.linear_model.LogisticRegression — scikit-learn 1.1.3 documentation
https://scikit-learn.org/stable/modules/linear_model.html#logistic-regression)

Метод логистической регрессии (также называемый логит-регрессией, logit regression) является методом классификации, несмотря на свое название. Логистическая регрессия возвращает вероятность того, что образец принадлежит к определенному классу, т.е. при предсказании возвращается вещественное число (от 0 до 1), а не только ответ (да/нет). Метод используют в задачах, в которых важен не только ответ, но и еще оценка вероятности, например, будет возвращен кредит, пойдет дождь или риск заболевания болезнью с вероятностью такой-то.

Постановка задачи такая же, как и для всех остальных линейных моделей. У нас есть обучающая выборка $X = (x_i, y_i)_{i=1}^{l}$, где $x_i \in R^n$ – набор из l объектов, у каждого такого объекта x_i есть n признаков $f_j(x_i)$ $(j = 1..n)$, известные ответы обучающей выборки – это один из двух классов $y_i \in \{-1, +1\}$.

В линейной модели **классификации** решение ищется в виде функции знака числа (sign) от суммы всех признаков $f_j(x)$ с коэффициентами w_j:

$$a(x, w) = \operatorname{sign}\langle x, w\rangle = \operatorname{sign}\left(\sum_{j=1}^{n} f_j(x) w_j\right).$$

В этой задаче определить функцию потерь для каждой точки x_i можно как пороговую (количество ошибок):

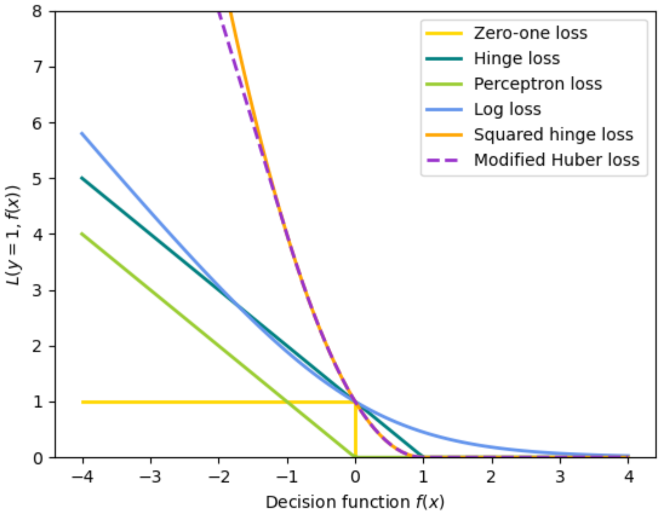
$$L(a, y) = [\langle x_i, w\rangle y_i < 0].$$

Вместо пороговой функции потерь используют ее непрерывные аппроксимации:

$$L(a, y) = [\langle x_i, w\rangle y_i < 0] \le L(\langle x_i, w\rangle y_i),$$

где $M_i(w) = (\langle x_i, w\rangle) y_i$ – **отступ** (margin) объекта x_i. При положительном отступе нет ошибки, при отрицательном отступе ошибка есть.

И в случае **метода логистической регрессии** выбирается функция потерь $L(t) = \log_2(1 + e^{-t})$. На рисунке она отмечена синим цветом (log loss). График ее близок к графику функции потерь шарнирной функции для метода опорных векторов, но она является дифференцируемой. Она также как и шарнирная функция потерь штрафует и за правильные ответы, тем больше, чем ближе точка к границе классов.



И также как и в остальных методах просуммировав ее для всех точек, получим функционал качества:

$$Q(w) = \sum_{i=1}^{l} L(\langle x_i, w \rangle y_i) = \sum_{i=1}^{l} \log_2(1 + e^{-\langle x_i, w \rangle y_i}) \to \min_w.$$

Также как и для других методов для борьбы с увеличением весов применяют **регуляризацию** L_2 (сокращает веса линейно зависимых признаков):

$$Q(w) = \sum_{i=1}^{l} L(\langle x_i, w \rangle y_i) + \tau \sum_{j=1}^{n} w_j^2 \to \min_w$$

или регуляризацию L_1 (обнуляет веса неинформативных признаков):

$$Q(w) = \sum_{i=1}^{l} L(\langle x_i, w \rangle y_i) + \tau \sum_{j=1}^{n} |w_i| \to \min_w$$

или их комбинацию (ElasticNet):

$$Q(w) = \sum_{i=1}^{l} L(\langle x_i, w \rangle y_i) + \tau_1 \sum_{j=1}^{n} w_j^2 + \tau_2 \sum_{j=1}^{n} |w_i| \to \min_w$$

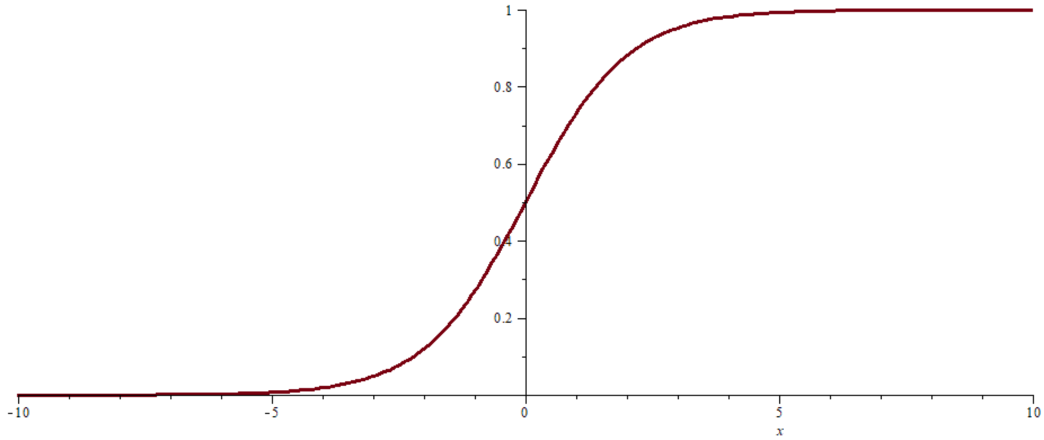
Находить минимум можно с помощью разных вариантов метода градиентного спуска, например стохастическим градиентом.

Кроме определения к какому классу относится точка по формуле $a(x, w) = \operatorname{sign} \langle x, w \rangle$, логистическая регрессионная модель оценивает еще **вероятность** принадлежности точки x к классу $y = 1$ по формуле:

$$p(y = 1 \mid x; w) = \sigma(\langle x, w \rangle),$$

где $\sigma(t)$ – это **логистическая сигмоидальная функция**:

$$\sigma(t) = \frac{1}{1 + e^{-t}},$$



которая любое вещественное значение переводит в отрезок [0, 1]. Если больше нуля $\langle x, w\rangle > 0$, т.е. метод относит точку к классу +1, то вероятность $p > 0.5$ и растет до 1 с увеличением расстояния от границы. И, наоборот, если $\langle x, w\rangle$ меньше нуля, т.е. метод относит точку к классу –1, то вероятность $p < 0.5$.

Вероятность принадлежности к классу 0 определяется:

$$p(y = 0 \mid x; w) = 1 - p(y = 1 \mid x; w) = 1 - \sigma(\langle x, w\rangle) = \sigma(-\langle x, w\rangle),$$

так чтобы сумма вероятностей для двух классов равнялась 1.

Обоснование функции потерь для логистической регрессии. Если предполагать, что наша выборка есть выборка независимых наблюдений $X = (x_i, y_i)_{i=1}^{l}$ с плотностью распределения $p(y \mid x;\ w)$, то из принципа максимального правдоподобия требуем, чтобы произведение вероятностей для отдельных точек было максимальным:

$$L(w) = \prod_{i=1}^{l} p(y_i \mid x_i; w) \to \max_{w}$$

Удобней искать максимум не произведения, а суммы, поэтому прологарифмируем эту функцию и перейдем:

$$L(w) = \log \prod_{i=1}^{l} p(y_i \mid x_i; w) = \sum_{i=1}^{l} \log p(y_i \mid x_i; w) \to \max_{w}$$

Вероятность для одной точки можно посчитать как сигмоиду от отступа:

$$p(y_i \mid x_i; w) = \frac{y+1}{2}\sigma(\langle x_i, w\rangle) + \frac{y-1}{2}\sigma(-\langle x_i, w\rangle) =$$

$$= \sigma(\langle x_i, w\rangle y_i) = \frac{1}{1 + e^{-\langle x_i, w\rangle y_i}}.$$

И тогда

$$L(w) = \sum_{i=1}^{l} \log \frac{1}{1 + e^{-\langle x_i, w\rangle y_i}} = -\sum_{i=1}^{l} \log(1 + e^{-\langle x_i, w\rangle y_i}) = \to \max_{w}$$

и можно от поиска максимума перейти к поиску минимума:

$$L(w) = \sum_{i=1}^{l} \log(1 + e^{-\langle x_i, w \rangle y_i}) = \to \min_w$$

И мы получили минимизацию того же функционала.

II. В библиотеке **Scikit-Learn** реализован метод **логистической регрессии** в классе sklearn.linear_model.LogisticRegression (sklearn.linear_model.LogisticRegression — scikit-learn 1.1.3 documentation) и sklearn.linear_model.LogisticRegressionCV (sklearn.linear_model.LogisticRegressionCV — scikit-learn 1.1.3 documentation). Описание метода в документации пакета Scikit-Learn можно посмотреть 1.1. Linear Models — scikit-learn 1.1.3 documentation.

Задания

I. Прочитайте пункт “Метод логистической регрессии” и рекомендуемую литературу и ответьте на вопросы:

1) Чем метод отличается от других линейных методов?
2) По какой формуле при предсказании вычисляется вероятность?
3) Какие основные параметры метода в библиотеке Scikit-Learn?
4) Какие преимущества и недостатки метода?

II. Выполните задание из в файла " statement-logistic.pdf".

Литература

[1] Рашка Себастьян, Мирджалили Вахид Python и машинное обучение: машинное и глубокое обучение с использованием Python, scikit-learn и TensorFlow 2, 3-е изд.: Пер. с англ. СПб. : ООО "Диалектика", 2020. 848 с.

[2] Жерон, Орельен. Прикладное машинное обучение с помощью Scikit-Learn и TensorFlow: концепции, инструменты и техники для создания интеллектуальных систем. Пер. с англ. СпБ.: ООО Альфа-книга: 2018. 688 с.

[3] К.В. Воронцов Математические методы обучения по прецедентам (теория обучения машин). 141 с. (Voron-ML-1.pdf)

[4] Машинное обучение (курс лекций, К.В.Воронцов): Логистическая регрессия (machinelearning.ru).
“Машинное обучение. Линейные методы. К.В. Воронцов, Школа анализа данных, Яндекс”: https://youtu.be/QIktmPA8nb0?t=3395
“Линейные методы классификации. Логистическая регрессия”: https://www.youtube.com/watch?v=VBmfuf11N-M)

[5] Крис Элбон Машинное обучение с использованием Python. Сборник рецептов. Пер. с англ. СПб. : БХВ-Петербург, 2019. 384 с.