Эмпирические байесовские нейронные сети

Басов Дмитрий Константинович

Аннотация

Данная статья посвящена применению техники эмпирического Байеса к байесовским нейронным сетям. Концептуально идея следующая:

1. Мы используем диагональное нормальное распределение для аппроксимации апостериорного распределения весов модели — $q(W)$.
2. Априорное распределение весов модели так же задаётся диагональным распределением с нулевым матожиданием — $p(W)$.
3. Используя вариационный вывод, мы приходим к ситуации, когда ELBO зависит от $KL(q(W) \mid\mid p(W))$. Так как оба распределения являются нормальными, то KL дивергенция считается аналитически.
4. Мотивация следующего этапа была взята из RVM — взять дисперсию априорного распределения весов модели $p(W)$ из данных. Там несложно берётся производная и всё получается красиво, кроме возможного деления $0/0$. Но сделав замену переменных, от этой беды можно уйти.

Пункты 1–3 в принципе были описаны в статье Weight Uncertainty in Neural Networks. А вот четвёртый пункт я ни в книгах, ни в статьях не находил.

1 Обозначения и сокращения

$N(\mu, \sigma^2)$ — нормальное распределение

$\boldsymbol{x} \odot \boldsymbol{y}$ — поэлементное произведение (произведение Адамара) векторов

$\mathcal{L}$ — Evidence Lower Bound (ELBO)

$KL(q \mid\mid p) = \int q(\boldsymbol{Z}) \cdot \ln \dfrac{q(\boldsymbol{Z})}{p(\boldsymbol{Z})} d\boldsymbol{Z}$ — дивергенция Кульбака–Лейблера

$\boldsymbol{x}$ — вектор признаков

$\boldsymbol{y}$ — вектор целевой переменной

D — датасет — пары значений $\{\boldsymbol{x_i}, \boldsymbol{y_i}\}$, где $i = 1, \ldots, L$

$\boldsymbol{W}$ — веса модели — случайная величина размерности M

$p(D|\boldsymbol{W}) = \prod_{i=1}^{L} p(\boldsymbol{y_i}|\boldsymbol{x_i}, \boldsymbol{W})$ — правдоподобие (likelihood)

$p(\boldsymbol{W})$ — априорное распределение весов модели (prior)

$p(\boldsymbol{W}|D)$ — апостериорное распределение весов модели (posterior)

$p(D)$ — маргинальная вероятность датасета (evidence)

$p(\boldsymbol{W}, D) = p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W}) = p(\boldsymbol{W}|D) \cdot p(D)$ — совместная вероятность весов модели и данных

$q(\boldsymbol{W}|\boldsymbol{\theta})$ — аппроксимация апостериорного распределения весов модели

$\boldsymbol{\theta}$ — обучаемые параметры байесовской модели

2 Введение

В классическом машинном обучении делается следующее предположение: веса модели $\boldsymbol{W}$ являются пусть и неизвестной, но фиксированной величиной. В этом случае можно получить точечную оценку весов модели согласно гипотезе максимального правдоподобия.

$$\boldsymbol{W_{ML}} = \operatorname*{argmax}_{\boldsymbol{W}} p(D|\boldsymbol{W})$$

Тогда распределение $p(\boldsymbol{y}|\boldsymbol{x}, D)$ аппроксимируется следующим образом:

$$p(\boldsymbol{y}|\boldsymbol{x}, D) \approx p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{W_{ML}})$$

Однако это справедливо при условии, что количество объектов в датасете D сильно больше количества весов модели ($L \gg M$). Если это не так, веса модели $\boldsymbol{W}$ могут слишком сильно подстроиться под обучающую выборку D, что черевато переобучением.

Для борьбы с переобучением используется ряд приёмов (штрафы на норму весов, early stopping, dropout), однако для их настройки требуются вычислительные ресурсы и отложенные (не участвующие в обучении) выборки данных.

Альтернативным подходом к машинному обучению является нахождение апостериорного распределения весов модели $p(\boldsymbol{W}|D)$ по теореме Байеса.

$$p(\boldsymbol{W}|D) = \frac{p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})}{\int p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W}) d\boldsymbol{W}}$$

Тогда предсказательное распределение $p(\boldsymbol{y}|\boldsymbol{x}, D)$ рассчитывается следующим образом:

$$p(\boldsymbol{y}|\boldsymbol{x}, D) = \int p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{W}) \cdot p(\boldsymbol{W}|D) d\boldsymbol{W} \approx \frac{1}{T} \sum_{t=1}^{T} p(\boldsymbol{y}|\boldsymbol{x}, \hat{\boldsymbol{W}}_{\boldsymbol{t}})$$

где $\hat{\boldsymbol{W}}_{\boldsymbol{t}}$ — сэмпл весов модели из $p(\boldsymbol{W}|D)$.

Байесовские модели машинного обучения можно рассматривать как ансамбль из бесконечного числа моделей, веса которых сэмплируются из распределения $p(\boldsymbol{W}|D)$. Такой подход устойчив к переобучению, так как о размере обучающей выборки не делается никаких предположений. Однако возникают следующие проблемы: выбор подходящего априорного распределения $p(\boldsymbol{W})$ и вычисление апостериорного распределения $p(\boldsymbol{W}|D)$.

Неудачный выбор $p(\boldsymbol{W})$ может сильно ухудшить качество модели, а расчёт апостериорного распределения $p(\boldsymbol{W}|D)$ требует вычисления интеграла по всему пространству весов модели, что для нейронных сетей практически невозможно.

В данной статье предлагается следующий подход к решению этих проблем.

1. Применяя вариационный вывод, распределение $p(\boldsymbol{W}|D)$ аппроксимируется распределением $q(\boldsymbol{W}|\boldsymbol{\theta})$, и задача сводится к максимизации нижней вариационной границы $\mathcal{L}$ по параметрам $\boldsymbol{\theta}$.
2. Распределение $q(\boldsymbol{W}|\boldsymbol{\theta})$ задаётся в виде нормального распределения с диагональной матрицей ковариации. То есть каждый вес модели определяется двумя числами, которые определяют его математическое ожидание и дисперсию. Таким образом, количество обучаемых параметров относительно классической нейронной сети возрастает в 2 раза.
3. Так как распределение $q(\boldsymbol{W}|\boldsymbol{\theta})$ является нормальным, применяя трюк с репараметризацией, становится возможным использовать градиентные методы для максимизации $\mathcal{L}$.
4. Априорное распределение весов модели $p(\boldsymbol{W})$ задаётся в виде нормального распределения с нулевым матожиданием и диагональной матрицей ковариации. То есть мы вносим следующее априорное знание: веса модели находятся около нуля.
5. Для определения дисперсии априорного распределения весов модели $p(\boldsymbol{W})$ применяется техника эмпирического Байеса. То есть значение дисперсии $p(\boldsymbol{W})$ берётся из датасета D. Так как распределения $q(\boldsymbol{W}|\boldsymbol{\theta})$ и $p(\boldsymbol{W})$ являются нормальными, оптимальное значение дисперсии распределения $p(\boldsymbol{W})$ определяется аналитически. Такой подход обладает большой универсальностью, однако из–за этого теряется теоретическая устойчивость к переобучению.
6. Вводятся новые параметры $\boldsymbol{\gamma}$ и $\boldsymbol{\rho}$, через которые выражаются матожидание и дисперсия распределения $q(\boldsymbol{W}|\boldsymbol{\theta})$. Это нужно, чтобы избежать неопределенности деления $\frac{0}{0}$, которая может возникнуть из–за определения дисперсии распределения $p(\boldsymbol{W})$ из данных.

Эксперименты показали, что предлагаемый подход сохраняет гибкость классических нейронных сетей и делает их устойчивыми к переобучению.

3 Постановка задачи

Задача машинного обучения с учителем в вероятностной постановке формулируется следующим образом: получить распределение вероятностей $p(\boldsymbol{y}|\boldsymbol{x}, D)$ целевой переменной $\boldsymbol{y}$ для неразмеченных $\boldsymbol{x}$, используя информацию из датасета D. В случае параметрических моделей, которыми являются нейронные сети, информация из датасета D кодируется посредством весов модели $\boldsymbol{W}$. Сделаем следующие преобразования:

$$p(\boldsymbol{y}|\boldsymbol{x}, D) = \int p(\boldsymbol{y}, \boldsymbol{W}|\boldsymbol{x}, D)d\boldsymbol{W} = \int p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x}, D) \cdot p(\boldsymbol{W}|\boldsymbol{x}, D)d\boldsymbol{W} = \int p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x}) \cdot p(\boldsymbol{W}|D)d\boldsymbol{W}$$

Пояснения:

- $p(\boldsymbol{y}|\boldsymbol{x}, D) = \int p(\boldsymbol{y}, \boldsymbol{W}|\boldsymbol{x}, D)d\boldsymbol{W}$, так как для любых случайных величин $\boldsymbol{a}$ и $\boldsymbol{b}$ справедливо $p(\boldsymbol{a}) = \int p(\boldsymbol{a}, \boldsymbol{b})d\boldsymbol{b}$
- $p(\boldsymbol{y}, \boldsymbol{W}|\boldsymbol{x}, D) = p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x}, D) \cdot p(\boldsymbol{W}|\boldsymbol{x}, D)$, так как для любых случайных величин $\boldsymbol{a}$ и $\boldsymbol{b}$ справедливо $p(\boldsymbol{a}, \boldsymbol{b}) = p(\boldsymbol{a}|\boldsymbol{b}) \cdot p(\boldsymbol{b})$
- $p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x}, D) = p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x})$, так как вся информация из датасета D отражена в весах $\boldsymbol{W}$
- $p(\boldsymbol{W}|\boldsymbol{x}, D) = p(\boldsymbol{W}|D)$, так как веса модели $\boldsymbol{W}$ не зависят от неразмеченных $\boldsymbol{x}$, которых не было в датасете D.

Получим выражение для $p(\boldsymbol{W}|D)$, используя формулу Байеса:

$$p(\boldsymbol{W}|D) = \frac{p(\boldsymbol{W}, D)}{p(D)} = \frac{p(\boldsymbol{W}, D)}{\int p(\boldsymbol{W}, D)d\boldsymbol{W}} = \frac{p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})}{\int p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})d\boldsymbol{W}}$$

Для аппроксимации распределения ответов модели можно воспользоваться методом Монте–Карло. Идея следующая: сэмплируем конечное количество весов $\hat{\boldsymbol{W}}_{\boldsymbol{1}}, \ldots, \hat{\boldsymbol{W}}_{\boldsymbol{T}}$ из распределения $p(\boldsymbol{W}|D)$ и аппроксимируем распределение $p(\boldsymbol{y}|\boldsymbol{x}, D)$ следующим образом:

$$p(\boldsymbol{y}|\boldsymbol{x}, D) \approx \frac{1}{T}\sum_{t=1}^{T} p(\boldsymbol{y}|\boldsymbol{x}, \hat{\boldsymbol{W}}_{\boldsymbol{t}})$$

Однако для этого нужно иметь возможность сэмплировать из распределения $p(\boldsymbol{W}|D)$. Получить аналитическое решение интеграла $\int p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})d\boldsymbol{W}$ можно только в очень ограниченном числе случаев.

Существует возможность сэмплировать из $p(\boldsymbol{W}|D)$, используя методы Монте–Карло для марковских цепей (MCMC). Однако для больших датасетов и большого числа весов это практически невозможно.

Альтернативным подходом к решению такой задачи является вариационный вывод — аппроксимация распределения $p(\boldsymbol{W}|D)$ распределением $q(\boldsymbol{W}|\boldsymbol{\theta})$, из которого сэмплировать намного проще.

4 Вариационный вывод

Идея вариационного вывода — сведение задачи байесовского вывода к задаче максимизации нижней вариационной границы (ELBO) $\mathcal{L}$, которая для распределения $q(\boldsymbol{W}|\boldsymbol{\theta})$ записывается следующим образом:

$$\mathcal{L} = \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{p(\boldsymbol{W}, D)}{q(\boldsymbol{W}|\boldsymbol{\theta})}d\boldsymbol{W}$$

Покажем мотивацию максимизации ELBO. Запишем выражение для $KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \,||\, p(\boldsymbol{W}|D))$ и преобразуем его, используя тождество $p(\boldsymbol{W}, D) = p(\boldsymbol{W}|D) \cdot p(D)$:

$$KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \,||\, p(\boldsymbol{W}|D)) = \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{q(\boldsymbol{W}|\boldsymbol{\theta})}{p(\boldsymbol{W}|D)}d\boldsymbol{W} = \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{p(D) \cdot q(\boldsymbol{W}|\boldsymbol{\theta})}{p(\boldsymbol{W}, D)}d\boldsymbol{W} =$$

$$\ln p(D) \cdot \int q(\boldsymbol{W}|\boldsymbol{\theta}) d\boldsymbol{W} - \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{p(\boldsymbol{W}, D)}{q(\boldsymbol{W}|\boldsymbol{\theta})} d\boldsymbol{W} = \ln p(D) - \mathcal{L}$$

Так как $\ln p(D)$ не зависит от $\boldsymbol{\theta}$, максимизация $\mathcal{L}$ по параметрам $\boldsymbol{\theta}$ ведёт к минимизации $KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W}|D))$. Тем самым, при максимизации $\mathcal{L}$ распределение $q(\boldsymbol{W}|\boldsymbol{\theta})$ будет приближаться к распределению $p(\boldsymbol{W}|D)$.

Преобразуем выражение для $\mathcal{L}$, используя тождество $p(\boldsymbol{W}, D) = p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})$:

$$\mathcal{L} = \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{p(\boldsymbol{W}, D)}{q(\boldsymbol{W}|\boldsymbol{\theta})} d\boldsymbol{W} = \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{p(D|\boldsymbol{W}) \cdot p(\boldsymbol{W})}{q(\boldsymbol{W}|\boldsymbol{\theta})} d\boldsymbol{W} =$$

$$\int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln p(D|\boldsymbol{W}) d\boldsymbol{W} - \int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln \frac{q(\boldsymbol{W}|\boldsymbol{\theta})}{p(\boldsymbol{W})} d\boldsymbol{W} =$$

$$\int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln p(D|\boldsymbol{W}) d\boldsymbol{W} - KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W})) =$$

$$\int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \sum_{i=1}^{L} \ln p(\boldsymbol{y_i}|\boldsymbol{x_i}, \boldsymbol{W}) d\boldsymbol{W} - KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W}))$$

Так как в случае нейронной сети аналитически посчитать интеграл по всему пространству весов $\boldsymbol{W}$ не представляется возможным, воспользуемся следующей аппроксимацией для $p(\boldsymbol{y}|\boldsymbol{x}, D)$ и $\mathcal{L}$:

$$p(\boldsymbol{y}|\boldsymbol{x}, D) \approx \int p(\boldsymbol{y}|\boldsymbol{W}, \boldsymbol{x}) \cdot q(\boldsymbol{W}|\boldsymbol{\theta}) d\boldsymbol{W} \approx \frac{1}{T} \sum_{t=1}^{T} p(\boldsymbol{y}|\boldsymbol{x}, \hat{\boldsymbol{W}}_{\boldsymbol{t}})$$

$$\mathcal{L} \approx \frac{1}{S} \sum_{j=1}^{S} \sum_{i=1}^{L} \ln p(\boldsymbol{y_i}|\boldsymbol{x_i}, \hat{\boldsymbol{W}}_{\boldsymbol{ij}}) - KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W}))$$

где $\hat{\boldsymbol{W}}_{\boldsymbol{t}}$ и $\hat{\boldsymbol{W}}_{\boldsymbol{ij}}$ — сэмплы весов модели из распределения $q(\boldsymbol{W}|\boldsymbol{\theta})$.

5 Задание функциональных форм распределений

Для дальнейшнего вывода положим, что распределения $p(\boldsymbol{W})$ и $q(\boldsymbol{W}|\boldsymbol{\theta})$ являются нормальными с диагональными матрицами ковариации:

$$q(\boldsymbol{W}|\boldsymbol{\theta}) = N(\boldsymbol{W}|\boldsymbol{\mu}, diag(\boldsymbol{\sigma_{q(W)}})^2)$$

$$p(\boldsymbol{W}) = N(\boldsymbol{W}|\boldsymbol{0}, diag(\boldsymbol{\sigma_{p(W)}})^2)$$

Так как распределение $q(\boldsymbol{W}|\boldsymbol{\theta})$ нормальное, мы можем использовать трюк с репараметризацией при сэмплировании весов, что позволяет использовать градиентные методы для оптимизации: $\hat{\boldsymbol{W}}_{\boldsymbol{ij}} = \boldsymbol{\epsilon_{ij}} \odot \boldsymbol{\sigma_{q(W)}} + \boldsymbol{\mu}$, где $\boldsymbol{\epsilon_{ij}}$ — сэмпл из $N(\boldsymbol{0}, \boldsymbol{I})$.

Так как распределения $p(\boldsymbol{W})$ и $q(\boldsymbol{W}|\boldsymbol{\theta})$ нормальные, $KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W}))$ считается аналитически:

$$KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \;||\; p(\boldsymbol{W})) = \frac{1}{2} \sum_{k=1}^{M} (\frac{\sigma^2_{q(W)_k}}{\sigma^2_{p(W)_k}} + \frac{\mu_k^2}{\sigma^2_{p(W)_k}} - \ln \frac{\sigma^2_{q(W)_k}}{\sigma^2_{p(W)_k}} - 1)$$

Априорное распределение весов модели $p(\boldsymbol{W})$ имеет нулевое математическое ожидание (из соображений симметрии), и среднеквадратическое отклонение $\boldsymbol{\sigma_{p(W)}}$. В классическом байесовском выводе параметр $\boldsymbol{\sigma_{p(W)}}$ должен задаваться до начала обучения, то есть являться гиперпараметром. Однако мы можем воспользоваться техникой эмпирического Байеса, то есть определить параметр априорного распределения $\boldsymbol{\sigma_{p(W)}}$ из данных.

6 Эмпирический Байес

Пусть $\boldsymbol{\alpha} = diag(\boldsymbol{\sigma_{p(W)}})^{-2}$. Тогда выражение для $KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W}))$ будет иметь следующий вид:

$$KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W})) = \frac{1}{2}\sum_{k=1}^{M}(\alpha_k \cdot \sigma^2_{q(W)_k} + \alpha_k \cdot \mu_k^2 - (\ln \sigma^2_{q(W)_k} + \ln \alpha_k) - 1)$$

Так как в выражении $\mathcal{L}$ интеграл $\int q(\boldsymbol{W}|\boldsymbol{\theta}) \cdot \ln p(D|\boldsymbol{W}) d\boldsymbol{W}$ не зависит от параметров распределения $p(\boldsymbol{W})$, то:

$$\frac{\partial \mathcal{L}}{\partial \alpha_k} = -\frac{\partial (KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W})))}{\partial \alpha_k} = -\frac{1}{2}(\sigma^2_{q(W)_k} + \mu_k^2 - \frac{1}{\alpha_k}) = -\frac{1}{2}(\sigma^2_{q(W)_k} + \mu_k^2 - \sigma^2_{p(W)_k})$$

Приравняв производную к нулю, получим:

$$-\frac{1}{2}(\sigma^2_{q(W)_k} + \mu_k^2 - \sigma^2_{p(W)_k}) = 0$$

$$\sigma^2_{p(W)_k} = \sigma^2_{q(W)_k} + \mu_k^2$$

Подставив полученное выражение для априорной дисперсии в $KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W}))$, получим:

$$KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W})) = \frac{1}{2}\sum_{k=1}^{M}\ln(1 + \frac{\mu_k^2}{\sigma^2_{q(W)_k}})$$

Таким образом, мы свели задачу к максимизации $\mathcal{L}$ по параметрам $\boldsymbol{\mu}$ и $\boldsymbol{\sigma_{q(W)}}$.

7 Замена переменных

При максимизации $\mathcal{L}$ могут возникнуть ситуации, когда какой–либо вес модели перестает быть случайной величиной и вырождается в ноль ($\sigma_{q(W)_k} \to 0$ и $\mu_k \to 0$). Это приведет к неопределенности деления 0 на 0 при вычислении KL–дивергенции.

Так же при градиентной оптимизации компоненты вектора $\boldsymbol{\sigma_{q(W)}}$ могут попасть в отрицательную область, что нежелательно, так как среднеквадратическое отклонение не может быть отрицательным по определению.

Чтобы избежать этих проблем, сделаем следующую замену переменных:

$$\boldsymbol{\sigma_{q(W)}} = \ln(1 + e^{\boldsymbol{\rho}}) = Softplus(\boldsymbol{\rho})$$

$$\boldsymbol{\mu} = \boldsymbol{\gamma} \odot \boldsymbol{\sigma_{q(W)}} = \boldsymbol{\gamma} \odot Softplus(\boldsymbol{\rho})$$

Тогда:

$$KL(q(\boldsymbol{W}|\boldsymbol{\theta}) \mid\mid p(\boldsymbol{W})) = \frac{1}{2}\sum_{k=1}^{M}\ln(1 + \frac{\mu_k^2}{\sigma^2_{q(W)_k}}) = \frac{1}{2}\sum_{k=1}^{M}\ln(1 + \gamma_k^2)$$

Таким образом, задача сводится к минимизации следующей функции потерь по параметрам $\boldsymbol{\rho}$ и $\boldsymbol{\gamma}$:

$$loss(\boldsymbol{\rho}, \boldsymbol{\gamma}) = -\frac{\mathcal{L}}{L} \approx -\frac{1}{S \cdot L}\sum_{j=1}^{S}\sum_{i=1}^{L}\ln p(\boldsymbol{y_i}|\boldsymbol{x_i}, \hat{\boldsymbol{W}}_{\boldsymbol{ij}}) + \frac{KL}{L}$$

где:

$$\hat{\boldsymbol{W}}_{\boldsymbol{ij}} = \boldsymbol{\epsilon} \odot \boldsymbol{\sigma} + \boldsymbol{\mu}$$

$$\boldsymbol{\epsilon} \sim N(\boldsymbol{0}, \boldsymbol{I})$$

$$\boldsymbol{\sigma} = Softplus(\boldsymbol{\rho})$$

$$\boldsymbol{\mu} = \boldsymbol{\gamma} \odot \boldsymbol{\sigma}$$

$$KL = \frac{1}{2}\sum_{k=1}^{M}\ln(1 + \gamma_k^2)$$

8 Алгоритм обучения

Задаем шаг градиентного спуска α и инициализируем параметры распределения $q(\boldsymbol{W}|\boldsymbol{\theta})$ — $\boldsymbol{\rho}$ и $\boldsymbol{\gamma}$. Затем повторяем, пока не достигнем критерия остановки:

1. $\boldsymbol{\sigma} \leftarrow Softplus(\boldsymbol{\rho})$ — расчёт среднеквадратических отклонений весов
2. $\boldsymbol{\mu} \leftarrow \boldsymbol{\gamma} \odot \boldsymbol{\sigma}$ — расчёт математических ожиданий весов
3. $\boldsymbol{\epsilon} \leftarrow N(0, 1)$ — сэмплирование случайных весов
4. $\hat{\boldsymbol{W}} \leftarrow \boldsymbol{\epsilon} \odot \boldsymbol{\sigma} + \boldsymbol{\mu}$ — репараметризация
5. $nll \leftarrow -\frac{1}{L}\sum_{i=1}^{L} \ln p(\boldsymbol{y_i}|\boldsymbol{x_i}, \hat{\boldsymbol{W}})$ — расчёт среднего отрицательного логарифма правдоподобия (возможна аппроксимация по батчам)
6. $kl \leftarrow \frac{1}{2}\sum_{k=1}^{M} \ln(1 + \gamma_k^2)$ — расчёт KL–дивергенции
7. $l \leftarrow nll + \frac{kl}{L}$ — расчёт функции потерь
8. $\boldsymbol{\rho} \leftarrow \boldsymbol{\rho} - \alpha\frac{\partial l}{\partial \boldsymbol{\rho}}$ — обновление $\boldsymbol{\rho}$
9. $\boldsymbol{\gamma} \leftarrow \boldsymbol{\gamma} - \alpha\frac{\partial l}{\partial \boldsymbol{\gamma}}$ — обновление $\boldsymbol{\gamma}$

9 Эксперименты

Для проверки своей гипотезы я выбрал Alzheimer's Disease Dataset. Данные были разбиты на тренировочную и тестовую часть в пропорции 80 на 20. В качестве архитектуры была выбрана полносвязная нейронная сеть с одним скрытым слоем и функцией активации ReLU. То есть:

$z = ReLU(matmul(x, W_1))$

$y = Sigmoid(matmul(z, W_2))$

Размерность скрытого состояния z варьировалась от 1 до 60. Для каждой размерности обучались 2 модели - классическая (без регуляризации) и байесовская. Для каждой модели производилась оценка ROC–AUC на тренировочной и тестовой выборках. На рисунке 1 представлены результаты экспериментов

10 Выводы

По результатам работы можно сделать следующие выводы:

- с ростом сложности модели байесовская нейронная сеть не переобучилась;
- значение ROC-AUC на тестовой выборке имеет очень высокую корреляцию со значением ROC-AUC на тренировочной выборке (0.97 по Пирсону). Следовательно, для подбора гиперпараметров можно ориентироваться на метрики, полученные по тренировочной выборке. Это даёт нам возможность отказаться от деления на тренировочную и валидационную выборки для подбора гиперпараметров.

Так же стоит отметить, что данный подход переносится на другие архитектуры нейронных сетей (рекуррентные, свёрточные, трансформеры).

Имплементация данного подхода была выполнена с использованием PyTorch. Весь исходный код для проведения экспериментов размещён по адресу https://github.com/dimabasow/bayesian-neural-networks.

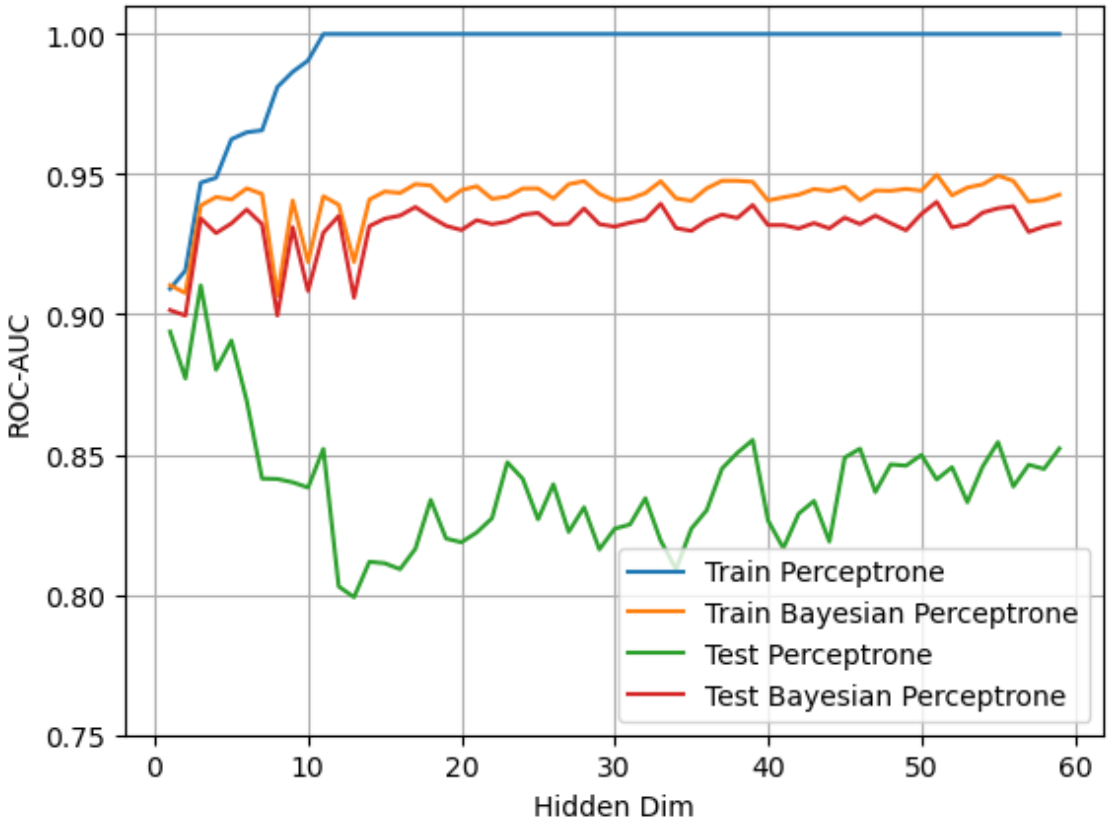

Рис. 1: Зависимость ROC–AUC от размерности скрытого состояния на тренировочных и тестовых данных