

Оглавление

DIAGNOSING BIAS VS VARIANCE	1
REGULARIZATION AND BIAS VARIANCE	3
LEARNING CURVES.....	6
DECIDING WHAT TO DO NEXT REVISITED	11

DIAGNOSING BIAS VS VARIANCE

0:00

Если вы запустите алгоритм обучения, и он не делает столько, сколько вы надеетесь, почти все время, это будет связано с тем, что у вас есть либо проблема с высоким уровнем смещения, либо проблема с большими отклонениями, другими словами, проблема недоучета или проблема переобучения. В этом случае очень важно выяснить, какая из этих двух проблем - это смещение или дисперсия или немного того, что у вас есть на самом деле. Поскольку знание того, какая из этих двух вещей происходит, даст очень сильный показатель того, полезны ли и перспективные способы улучшить ваш алгоритм. В этом видео я хотел бы углубиться в эту проблему смещения и дисперсии и лучше понять их, поскольку выяснял, как смотреть в алгоритме обучения, оценивать или диагностировать, может ли быть проблема смещения или проблема дисперсии, поскольку это будет иметь решающее значение для выяснения того, как улучшить производительность алгоритма обучения, который вы будете внедрять. Итак, вы уже видели эту цифру несколько раз, когда, если вы согласитесь на две простые гипотезы, такие как прямая линия, которая подкрепляет данные, если вы вписываете две сложные гипотезы, то это может идеально соответствовать набору тренировок, но перегружать данные, и это может быть гипотезой некоторого промежуточного уровня сложности некоторых, возможно, двухмерных многочленов степени или не слишком низкой и не слишком высокой степени, которая как раз и дает наилучшую ошибку обобщения по этим параметрам. Теперь, когда мы вооружены понятием цепной тренировки и проверки в наборах тестов, мы можем понять концепции смещения и дисперсии немного лучше. Конкретно, допустим, чтобы наша ошибка обучения и ошибка перекрестной проверки были определены как в предыдущих видео. Просто скажите квадратную ошибку, среднеквадратичную ошибку, измеренную на наборах тренировок или измеренную по набору перекрестной проверки. Теперь давайте построим следующий рисунок. На горизонтальной оси я собираюсь рассчитать степень полинома. Поэтому, когда я иду вправо, я собираюсь установить полиномы высшего и высшего порядка. Итак, где слева от этой фигуры, где возможно d равно единице, мы собираемся установить очень простые функции, тогда как мы здесь справа от горизонтальной оси, у меня гораздо больше значений d , многочлена многочлена, Таким образом, это будет соответствовать установке более сложных функций для вашего набора тренировок. Давайте посмотрим на ошибку обучения и ошибку перекрестной проверки и запишите их на этом рисунке. Начнем с ошибки обучения. По мере увеличения степени полинома мы сможем лучше и лучше подогнать наш тренировочный комплект, и если d будет равным единице, тогда будет высокая ошибка в обучении, если у нас будет очень высокая степень полинома. Наша обучающая ошибка будет очень низким, может быть, даже 0, потому что он будет соответствовать тренировочному набору очень хорошо. Таким образом, по мере увеличения степени полинома, обычно обнаруживается,

что ошибка обучения уменьшается. Поэтому я собираюсь написать J-точный трент там, потому что наша обучающая ошибка имеет тенденцию уменьшаться со степенью полинома, который мы подходим к данным. Далее, давайте посмотрим на ошибку перекрестной проверки или если это произойдет, если мы посмотрим на ошибку набора тестов, мы получим довольно похожий результат, как если бы мы построили ошибку перекрестной проверки. Таким образом, мы знаем, что если d равно единице, мы подходим к очень простой функции, и поэтому мы можем быть подкреплены набором тренировок, и поэтому это будет очень высокая ошибка перекрестной проверки. Если мы поместим промежуточный полином промежуточной степени, в нашем примере на предыдущем слайде мы имеем d , равный двум, у нас будет гораздо более низкая ошибка перекрестной проверки, потому что мы находим гораздо более подходящую для данных. И наоборот, если d были слишком высокими. Поэтому, если d взял значение 4, то мы снова переработаем, и поэтому мы получим большое значение для ошибки перекрестной проверки. Итак, если бы вы плавно изменяли это положение и строили кривую, вы могли бы получить такую кривую, где это JCV тета. Опять же, если вы планируете J тест тета, вы получаете что-то очень похожее. Таким образом, этот вид сюжета также помогает нам лучше понимать понятия смещения и дисперсии. Конкретно, предположим, что вы применили алгоритм обучения, и он не работает так хорошо, как вы надеетесь, поэтому, если ваша ошибка с перекрестной проверкой или ваша ошибка набора тестов высока, как мы можем понять, если алгоритм обучения страдает от высокого уклона или страдают от большой дисперсии? Таким образом, установка ошибки перекрестной проверки является высокой, соответствует либо этому режиму, либо этому режиму. Таким образом, этот режим слева соответствует высокой проблеме смещения. То есть, если вы устанавливаете многочлен с более низким порядком, такой как d , равный единице, когда нам действительно нужен полином более высокого порядка для соответствия данным, тогда как в противоположность этому этот режим соответствует проблеме с большими отклонениями. То есть, если d степень полинома слишком велика для набора данных, который у нас есть, и эта цифра дает нам представление о том, как различать эти два случая. Конкретно, для случая с высоким уклоном, то есть в случае недофинансирования, мы обнаруживаем, что и ошибка перекрестной проверки, и ошибка обучения будут высокими. Итак, если ваш алгоритм страдает от проблемы смещения, ошибка набора обучения будет высокой, и вы можете обнаружить, что ошибка перекрестной проверки также будет высокой. Это может быть близко, может быть, чуть выше, чем ошибка обучения. Итак, если вы видите эту комбинацию, это признак того, что ваш алгоритм может страдать от большого уклона. Напротив, если ваш алгоритм страдает от высокой дисперсии, то, если вы посмотрите здесь, мы заметим, что поезд J, то есть ошибка обучения, будет низким. То есть, вы очень хорошо подходите к учебному набору, тогда как ваша ошибка перекрестной проверки, предполагающая, что это, скажем, квадратная ошибка, которую мы пытаемся свести к минимуму, в то время как, напротив, ваша ошибка при наборе перекрестной проверки или вашем кресте функция или набор перекрестных проверок будет намного больше, чем ваша ошибка набора тестов. Итак, это двойной знак больше. Это символ карты намного больше, чем два знака. Поэтому, если вы видите эту комбинацию значений, то это ключ к пониманию того, что ваш алгоритм обучения может страдать от большой дисперсии и может перерабатывать. Ключ, который отличает эти два случая, заключается в том, что если у вас есть проблема с высоким уровнем смещения, ваша ошибка набора обучения также будет высокой, так как ваша гипотеза просто не подходит для тренировочного набора. Если у вас проблема с высокой дисперсией, ваша ошибка набора тестов обычно будет низкой, что намного ниже вашей

ошибки перекрестной проверки. Поэтому, надеюсь, это дает вам несколько лучшее понимание двух проблем смещения и дисперсии. У меня еще много чего сказать о смещении и дисперсии в следующих нескольких видео, но то, что мы увидим позже, состоит в том, что, диагностируя, может ли алгоритм обучения страдать от высокого предвзятости или высокого отклонения, я покажу вам еще больше подробнее о том, как это сделать в последующих видеороликах. Но мы увидим, что, выясняя, может ли алгоритм обучения страдать от высокой предвзятости или высокой дисперсии или сочетания того и другого, то это даст нам гораздо лучшее руководство для того, что может быть многообещающим, чтобы попытаться улучшить производительность алгоритм обучения.

REGULARIZATION AND BIAS VARIANCE

0:00

Вы видели, как регуляризация может помочь предотвратить чрезмерную установку. Но как это влияет на смещение и отклонения алгоритма обучения? В этом видео я хотел бы углубиться в вопрос о предвзятости и дисперсиях и рассказать о том, как он взаимодействует и зависит от регуляризации вашего алгоритма обучения.

0:22

Предположим, что мы устанавливаем высокий автоматический многочлен, как показано здесь, но чтобы предотвратить установку, нам нужно использовать регуляризацию, как показано здесь. Таким образом, у нас есть этот термин регуляризации, чтобы попытаться сохранить значения према до малого. И, как обычно, регуляризации происходит от $J = 1$ до m , а не от $j = 0$ до m . Рассмотрим три случая. Первый случай имеет очень большое значение параметра регуляризации лямбда, например, если лямбда была равна 10000. Некоторая огромная ценность.

0:54

В этом случае все эти параметры, θ_1 , θ_2 , θ_3 и т. д. Будут сильно оштрафованы, и поэтому мы заканчиваем тем, что большинство этих значений параметров ближе к нулю. И гипотеза будет примерно h от x , равная или приблизительно равная нулю. Таким образом, мы получаем гипотезу, которая более или менее выглядит так, более или менее плоская, постоянная прямая линия. И поэтому эта гипотеза имеет большое предвзятое отношение, и она плохо подходит для этого набора данных, поэтому горизонтальная прямая линия просто не очень хорошая модель для этого набора данных. С другой стороны, если мы имеем очень маленькое значение лямбда, например, если лямбда равна нулю. В этом случае, учитывая, что мы устанавливаем полином высокого порядка, это обычная переопределяющая настройка. В этом случае, учитывая, что мы подгоняем многочлен высокого порядка, в основном, без регуляризации или с очень минимальной регуляризацией, мы получаем нашу обычную высокую дисперсию, по сравнению с настройкой установки. Это в основном, если лямбда равна нулю, мы просто подходим к нашей регуляризации, чтобы это соответствовало гипотезе. И только в том случае, если у нас есть промежуточное значение больше, которое не является слишком большим и слишком маленьким, что мы получаем данные параметров, которые дают нам разумную форму для этих данных. Итак, как мы можем автоматически выбрать хорошее значение для параметра регуляризации?

2:19

Чтобы повторить, вот наша модель, и вот наша цель алгоритма обучения. Для настройки, в которой мы используем регуляризацию, позвольте мне определить, что $J_{\text{train}}(\theta)$ будет чем-то другим, быть целью оптимизации, но без термина регуляризации. Раньше, в более раннем видео, когда мы не использовали регуляризацию, я определяю J -последовательность данных как ту же, что и J_{θ} , как функцию причины, но когда мы используем регуляризацию, когда шесть ярусов под термином, которые мы собираемся определить J тренируйте мой тренировочный набор только в моей сумме квадратов ошибок на тренировочном наборе или моей средней квадратичной ошибки в обучающем наборе без учета этой регуляризации. И аналогичным образом я также собираюсь определить ошибку перекрестных проверок и проверить эту ошибку, как и раньше, как среднюю сумму квадратов ошибок при проверке перекрестных ссылок в наборах тестов, чтобы просто суммировать мои определения J_{train} J_{CU} и J - это всего лишь средний квадрат, который содержит половину другой квадратной записи на экзамене по тестированию тестового набора без дополнительного условия регуляризации. Таким образом, мы можем автоматически выбрать параметр регуляризации λ . Так что я обычно делаю, может быть, есть ряд значений λ , которые я хочу попробовать. Поэтому я мог бы подумать, что вы не используете регуляризацию, или вот несколько значений, которые я могу попробовать λ , учитывая $\lambda = 0.01, 0.02, 0.04$ и т. Д. И я обычно устанавливаю их в несколько раз, до тех пор, пока не может быть больше, если я сделаю это в кратных 2, и я получу 10.24. Это точно 10, но это достаточно близко. И от трех до четырех знаков после запятой не будет сильно влиять на ваш результат. Таким образом, это дает мне, может быть, 12 разных моделей. И я пытаюсь выбрать месяц, соответствующий 12 различным значениям регуляризации параметра λ . И, конечно, вы также можете перейти к значениям меньше 0,01 или значениям больше 10, но я просто усекал его здесь для удобства. Учитывая вопрос об этих 12 моделях, мы можем сделать следующее: мы можем взять эту первую модель с λ равной нулю и минимизировать мою стоимость J данных, и это даст мне некоторый параметр активных данных. И похоже на предыдущее видео, позвольте мне просто обозначить это как супер-сценарий.

4:49

И тогда я могу взять свою вторую модель с λ , установленной на 0.01, и свести к минимуму мою функцию стоимости теперь, используя λ , равную 0,01, конечно. Чтобы получить несколько разных параметров вектора θ . Обозначим, что $\theta(2)$. И для этого я заканчиваю $\theta(3)$. Так что, если часть для моей третьей модели. И так до тех пор, пока моя последняя модель с λ не будет установлена в 10 или 10.24, я в конечном итоге получаю эту $\theta(12)$. Затем я могу рассказать обо всех этих гипотезах, обо всех этих параметрах и использовать мою проверку перекрестных ссылок, чтобы проверить их, чтобы я мог посмотреть на мою первую модель, вторую модель, соответствующую этим различным значениям параметра регуляризации, и оценить их с помощью моей перекрестной проверки основана на измерении средней квадратичной ошибки каждого из этих квадратных векторных параметров θ на моих наборах проверки перекрестных ссылок. И тогда я бы выбрал то, что одна из этих 12 моделей дает мне самую низкую ошибку в наборе проверки трансляции. И, скажем, ради этого примера, я в конечном итоге собираю θ_5 , полином 5-го порядка, потому что это имеет самую низкую ошибку проверки причины. Сделав это, наконец, что бы я сделал, если бы хотел сообщить о каждой ошибке тестового набора, нужно взять параметр θ_5 , который я выбрал, и посмотреть,

насколько хорошо он работает на моем тестовом наборе. Итак, еще раз, вот как будто мы применили этот параметр, θ , к моему набору кросс-валидации, поэтому я откладываю отдельный тестовый набор, который я собираюсь использовать, чтобы лучше оценить, как хорошо, мой вектор параметров, θ , будет обобщен на ранее невидимые примеры. Таким образом, выбор модели применяется для выбора параметра регуляризации λ . Последнее, что я хотел бы сделать в этом видео, - получить лучшее представление о том, как меняется перекрестная проверка и ошибка обучения, поскольку мы меняем параметр регуляризации λ . И это просто напоминание, это была наша первоначальная стоимость на θ . Но для этого мы собираемся определить ошибку обучения без использования параметра регуляризации и перекрестной проверки, не используя параметр регуляризации.

7:07

И то, что я хотел бы сделать, это заговорить этот J_{train} и заговорить этот J_{cv} , а это значит, насколько хорошо моя гипотеза делает на тренировочном наборе и как моя гипотеза делает, когда она пересекает проверки. Поскольку я изменяю свой параметр регуляризации λ .

7:27

Так, как мы видели раньше, если λ мала, то мы не используем много регуляризации

7:35

и мы сталкиваемся с большим риском более подходящего, тогда как если λ велика, то есть, если бы мы были на правой части этой горизонтальной оси, тогда, при большом значении λ , мы подвергаемся более высокому риску иметь предвзятую проблему, поэтому, если вы сюжет J_{train} и J_{cv} , вы обнаружите, что для небольших значений λ вы можете поместиться в торговый набор относительно, потому что вы не регулируете. Таким образом, при небольших значениях λ термин регуляризации в основном уходит, и вы просто сворачиваете почти просто серые стрелки. Поэтому, когда λ мала, вы получаете небольшую ценность для J_{train} , тогда как если λ велика, тогда у вас есть проблема с высоким уровнем смещения, и вы можете не чувствовать себя хорошо в этом обучении, так что вы в конечном итоге получаете ценность. Таким образом, J_{train} of θ будет стремиться увеличиваться, когда λ увеличивается, потому что большое значение λ соответствует высокому смещению, где вы, возможно, даже не подходите к своим тренировкам, в то время как небольшое значение λ соответствует, если вы действительно можете подходить очень высоко допустим, полином по вашим данным, скажем так. После ошибки проверки стоимости мы получим такую цифру,

8:51

где здесь справа, если у нас есть большое значение λ , мы можем оказаться в подгонке, и поэтому это режим смещения. И поэтому ошибка перекрестной проверки будет высокой. Позвольте мне оставить все это на этом J_{cv} (θ), потому что так, с большим уклоном, мы не будем подходить, мы не будем хорошо работать в наборах перекрестной проверки, тогда как здесь слева это высокая дисперсия режим, где мы имеем два меньших значения с более длинными, то мы можем быть более подходящими для данных. И поэтому, поместив данные, ошибка перекрестной проверки также будет высокой. Итак, это ошибка перекрестного подтверждения и то, что торговая ошибка может выглядеть на торговой позиции, поскольку мы меняем параметр регуляризации λ . И так еще раз, часто будет какое-то промежуточное

значение лямбда, которое является правильным или лучше всего работает с точки зрения наличия небольшой ошибки перекрестной проверки или небольшой тестовой тета. И в то время как кривые, которые я здесь нарисовал, несколько мультяшны и несколько идеализированы, поэтому на реальных наборах данных кривые, которые вы получаете, могут оказаться немного более грязными и немного более шумными, чем это. Для некоторых наборов данных вы действительно увидите их для разных тенденций, и, посмотрев на график ошибки перекрестной проверки, вы можете либо вручную, либо попытаться выбрать точку, которая минимизирует ошибку перекрестной проверки и выберет значение лямбда что соответствует низкой ошибке перекрестной проверки. Когда я пытаюсь выбрать параметр регуляризации лямбда для алгоритма обучения, часто я нахожу, что построение фигуры, подобной приведенной здесь, помогает мне лучше понять, что происходит, и помогает мне проверить, что я действительно выбираю хорошее значение для параметра регуляризации монитор. Поэтому, надеюсь, это даст вам больше информации о регуляризации, и это повлияет на смещение и дисперсию алгоритма обучения. К настоящему времени вы заметили предвзятость и отклонения от множества разных точек зрения. И то, что мы хотели бы сделать в следующем видео, - это взять все идеи, которые мы прошли, и построить их, чтобы собрать диагностику, называемую кривыми обучения, которая является инструментом, который я часто использую для диагностики, если алгоритм обучения может быть страдают от проблемы смещения или проблемы с отклонениями, или немного того и другого.

LEARNING CURVES

0:00

В этом видео я хотел бы рассказать вам о кривых обучения.

0:03

Кривые обучения часто очень полезны для сюжета. Если вы хотите здравомыслия, проверьте, правильно ли работает ваш алгоритм, или если вы хотите улучшить производительность алгоритма.

0:13

И кривые обучения - это инструмент, который я на самом деле часто использую, чтобы попытаться диагностировать, может ли алгоритм физического обучения страдать от предвзятости, разновидности проблемы с отклонениями или немного того и другого.

0:27

Вот что такое кривая обучения. Для построения кривой обучения то, что я обычно делаю, - это график J , который, скажем,

0:35

средняя квадратная ошибка на моем обучающем наборе или J_{cv} , которая является средней квадратичной ошибкой в моем наборе проверки перекрестных ссылок. И я собираюсь заговорить, что как функция от m , это зависит от количества примеров обучения, которые у меня есть. И поэтому m обычно является постоянным, как, может быть, у меня есть, знаете, 100 учебных примеров, но то, что я собираюсь сделать, искусственно использует мое упражнение для тренировки. Таким образом, я намеренно ограничиваю себя использованием, скажем, 10 или 20

или 30 или 40 примеров обучения, и рассказываю, что такое ошибка обучения, и что такое перекрестная проверка для этих самых маленьких упражнений. Итак, давайте посмотрим, как выглядят эти сюжеты. Предположим, что у меня есть только один пример обучения, показанный здесь в этом первом примере, и предположим, что я подстраиваюсь к квадратичной функции. Ну, я

1:22

имеют только один пример обучения. Я собираюсь вписаться в это совершенно правильно? Вы знаете, просто установите квадратичную функцию. У меня будет 0 ошибка на одном примере обучения. Если у меня есть два примера обучения. Ну, квадратичная функция также может очень хорошо подходить. Так,

1:37

даже если я использую регуляризацию, я, вероятно, вполне подойду к этому. И если я не использую нервную регуляризацию, я точно подойду к этому, и если у меня снова будет три примера обучения. Да, я могу идеально подогнать квадратичную функцию, если m равно 1 или m равно 2 или m равно 3,

1:54

моя ошибка обучения в моем учебном наборе будет 0, если я не использую регуляризацию, или она может немного большей в 0, если я использую регуляризацию и, кстати, если у меня есть большой набор тренировок, и я искусственно ограничиваю размер моего тренировочного набора, чтобы тренировать J . Здесь, если я устанавливаю M равным 3, скажем, и я тренируюсь только по трем примерам, то для этой цифры я буду измерять мою ошибку обучения только по трем примерам, которые действительно соответствуют моим данным

2:27

и поэтому даже я должен сказать 100 примеров обучения, но если я хочу заговорить, что моя ошибка обучения равна m , равно 3. Что я собираюсь делать

2:34

заключается в том, чтобы измерить ошибку обучения на трех примерах, которые я действительно соответствовал моей гипотезе 2.

2:41

И не все другие примеры, которые я умышленно пропустил из учебного процесса. Поэтому просто для того, чтобы суммировать то, что мы видели, является то, что если размер набора для обучения невелик, ошибка обучения также будет небольшой. Потому что вы знаете, у нас есть небольшой набор для тренировок, который будет очень легко подогнать ваш тренировочный набор очень хорошо, может быть, даже сейчас прекрасно сказать, что у нас есть m , например, 4. Ну, тогда квадратичная функция может быть более подходящей для этого набора данных, и если у меня будет m равно 5, то вы знаете, может быть, квадратичная функция поместится так, чтобы оставаться там так, тогда как мой набор тренировок становится больше.

3:16

Становится все труднее и труднее гарантировать, что я могу найти квадратичную функцию, которая отлично обрабатывает все мои примеры. Таким образом, по мере того, как размер набора тренировок растет, вы обнаружите, что моя средняя ошибка обучения фактически увеличивается, и поэтому, если вы нарисуете эту цифру, то, что вы обнаружите, заключается в том, что ошибка набора учебных заданий, являющаяся средней ошибкой в вашей гипотезе, растет с ростом m и просто повторять, когда интуиция такова, что когда m мало, если у вас очень мало примеров обучения. Это очень легко подходит для каждого из ваших учебных примеров, и ваша ошибка будет небольшой, тогда как когда m будет больше, вам станет сложнее все учебные примеры, и поэтому ваша ошибка набора навыков станет более крупной, как насчет креста Ошибка проверки. Ну, перекрестная проверка - это моя ошибка в этом наборе перекрестной проверки, которого я не видел, и поэтому, знаете, когда у меня очень небольшой набор тренировок, я не собираюсь хорошо обобщать, просто не буду делать хорошо что. Итак, правильно, эта гипотеза здесь не выглядит хорошей, и только тогда, когда я получаю больший набор тренировок, вы знаете, я начинаю получать гипотезы, которые, возможно, подходят для данных несколько лучше. Таким образом, ваша ошибка перекрестной проверки и ваша ошибка набора тестов будут уменьшаться по мере увеличения вашего набора заданий, потому что чем больше данных у вас есть, тем лучше вы делаете это при обобщении на новые примеры. Итак, чем больше данных у вас есть, тем лучше ваша гипотеза. Так что если вы планируете поезд, и J_{cv} , это то, что вы получаете. Теперь давайте посмотрим, как выглядят кривые обучения, если у нас есть либо проблемы с высоким уровнем смещения, либо большие отклонения. Предположим, что ваша гипотеза имеет высокую предвзятость и объясняет это. Я собираюсь использовать пример, чтобы установить прямую линию к данным, которые, как вы знаете, не могут быть хорошо подогнаны по прямой.

5:09

Таким образом, мы получаем гипотезы, которые, возможно, выглядят так.

5:13

Теперь давайте подумаем, что произойдет, если мы будем увеличивать размер обучаемого набора. Поэтому, если вместо пяти примеров, как то, что я там нарисовал, представьте, что у нас есть намного больше примеров обучения.

5:25

Хорошо, что происходит, если вы подойдете к этой прямой. То, что вы находите, это то, что вы в конечном итоге знаете, почти такая же прямая линия. Я имею в виду прямую линию, которая просто не может соответствовать этим данным и получать больше тонны данных, поэтому прямая линия не изменится так сильно. Это наилучшая возможная прямая привязка к этим данным, но прямая линия просто не подходит для этого набора данных. Итак, если вы производите ошибку проверки,

5:49

это то, как он будет выглядеть.

5:51

Вариант слева, если у вас уже есть минимальный размер набора тренировок, как вы знаете, может быть, только один пример обучения, и он не преуспеет. Но к тому времени, когда вы достигли определенного количества примеров обучения, вы почти поместили наилучшую возможную прямую линию, и даже если у вас получится гораздо больший размер набора упражнений, гораздо большее значение m , вы знаете, в основном получая одну и ту же прямую линию, и, следовательно, ошибку перекрестной проверки - позвольте мне наметить это - или проверить ошибку набора или плато, или сгладить довольно скоро, как только вы достигнете определенного количества примеров обучения, если только вы в значительной степени подходите максимально возможная прямая линия. А как насчет ошибки в обучении? Ну, ошибка обучения снова будет небольшой.

6:34

И то, что вы находите в случае с высоким смещением, заключается в том, что ошибка обучения будет близка к ошибке проверки перекрестной связи, потому что у вас так мало параметров и столько данных, по крайней мере, когда m велико. Производительность на обучающем наборе и наборе кросс-проверки будет очень схожей.

6:53

Итак, вот как будут выглядеть ваши кривые обучения, если у вас есть алгоритм с высоким уклоном.

7:00

И, наконец, проблема с высоким уклоном отражается в том, что как ошибка перекрестной проверки, так и ошибка обучения высоки, и поэтому вы получаете относительно высокое значение как J_{cv} , так и J -го.

7:15

Это также подразумевает что-то очень интересное, то есть, если алгоритм обучения имеет высокую предвзятость, поскольку мы получаем все больше и больше примеров обучения, то есть по мере того, как мы переходим справа от этой цифры, мы заметим, что ошибка перекрестной проверки не идет много, он в основном откармливается, и поэтому, если алгоритмы обучения действительно страдают от высокой предвзятости.

7:36

Получение большего объема данных по обучению само по себе на самом деле не поможет, и, как наш пример на рисунке справа, здесь у нас было всего пять тренировок. примеры, и мы заполняем определенную прямую линию. И когда у нас было более тонких данных о тренировках, мы все же закончили примерно ту же прямую линию. И так, если алгоритм обучения имеет высокий уровень предвзятости, дайте мне намного больше данных о тренировках. Это фактически не поможет вам получить более низкую ошибку перекрестной проверки или ошибку набора тестов. Поэтому знание того, что ваш алгоритм обучения страдает от высокой предвзятости, кажется полезной вещью, потому что это может помешать вам тратить много времени на сбор дополнительных данных о тренировках, где это может просто не оказаться полезным. Затем рассмотрим настройку алгоритма обучения, который может иметь высокую дисперсию.

8:21

Давайте просто посмотрим на ошибку обучения вокруг, если у вас очень умная тренировка, подобная пяти примерам обучения, показанным на рисунке справа, и если мы подходим к утверждению полинома очень высокого порядка,

8:34

и я написал многочлен сотой степени, который на самом деле никто не использует, а просто иллюстрация.

8:39

И если мы используем довольно небольшое значение λ , возможно, не равное нулю, но довольно маленькое значение λ , то мы закончим, вы знаете, очень хорошо подобрав эти данные, используя функцию, которая накладывает эту надбавку. Итак, если размер набора для обучения невелик, наша обучающая ошибка, т. Е. Тренировка тета будет небольшой.

9:03

И поскольку этот размер набора обучения немного увеличивается, вы знаете, мы все еще немного перечитаем эти данные, но также становится немного сложнее подогнать этот набор данных, и поэтому, по мере увеличения размера набора для обучения, мы найдем что J_{train} увеличивается, потому что немного легче установить тренировочный набор, когда у нас есть больше примеров, но ошибка набора тренировок будет по-прежнему довольно низкой. Теперь, как насчет ошибки перекрестной проверки? Ну, в условиях высокой дисперсии гипотеза перерабатывает, поэтому ошибка перекрестной проверки остается высокой, даже, как мы узнаем, умеренное количество примеров обучения, и, возможно, ошибка перекрестной проверки может выглядеть так. И индикативная диагностика, что у нас есть проблема с большими отклонениями,

9:50

является тот факт, что существует большой разрыв между ошибкой обучения и ошибкой проверки перекрестных ссылок.

9:57

И глядя на эту фигуру. Если мы подумаем о добавлении дополнительных данных обучения, то есть, взяв эту фигуру и экстраполируя вправо, мы можем сказать, что вы знаете, что две кривые, синяя кривая и кривая пурпурного цвета сходятся друг к другу. Итак, если бы мы экстраполировали эту цифру вправо, то, похоже, вероятно, что ошибка обучения будет продолжать расти, и

10:27

ошибка перекрестной проверки будет продолжать снижаться. И то, что нам действительно волнует, это ошибка перекрестной проверки или ошибка набора тестов, верно? Таким образом, в таком виде мы можем сказать, что если мы продолжим добавлять примеры обучения и экстраполировать вправо, то наша ошибка перекрестной проверки будет продолжать снижаться. И, таким образом, при настройке с высокой дисперсией получение данных о тренировках действительно поможет. И снова это кажется полезной вещью, чтобы узнать, страдает ли ваш

алгоритм обучения высокой проблемой дисперсии, потому что это говорит вам, например, что вам может быть полезно узнать, можете ли вы пойти и получить дополнительную подготовку данные.

11:03

Теперь, на предыдущем слайде и этом слайде, я нарисовал довольно чистые довольно идеализированные кривые. Если вы нарисуете эти кривые для реального алгоритма обучения, иногда вы действительно увидите, как вы знаете, довольно много кривых, как то, что я здесь нарисовал. Хотя, иногда вы видите кривые, которые немного шумнее и немного грязнее, чем это. Но построение кривых обучения, подобных этим, может часто вам сообщать, часто может помочь вам понять, не страдает ли ваш алгоритм обучения предвзятостью или дисперсией, или даже немного того и другого. Поэтому, когда я пытаюсь улучшить производительность алгоритма обучения, одна вещь, которую я почти всегда делаю, заключается в построении этих кривых обучения, и обычно это даст вам лучшее представление о том, есть ли проблема смещения или отклонения.

11:44

И в следующем видео мы увидим, как это может помочь предложить конкретные действия - предпринять или не принять, чтобы попытаться повысить производительность вашего алгоритма обучения.

DECIDING WHAT TO DO NEXT REVISITED

0:00

Мы говорили о том, как оценивать алгоритмы обучения, говорили о выборе модели, много говорили о предвзятости и дисперсии. Итак, как это помогает нам понять, какие потенциально плодотворные, потенциально не полезные вещи, чтобы попытаться сделать так, чтобы улучшить производительность алгоритма обучения.

0:15

Давайте вернемся к нашему оригинальному мотивирующему примеру и подойдем к результату.

0:21

Итак, вот наш более ранний пример, возможно, имеющий правильную линейную регрессию и выяснение, что он не работает так хорошо, как мы надеемся. Мы сказали, что у нас есть это меню опций. Итак, есть ли способ выяснить, какие из них могут быть плодотворными? Во-первых, все это стало больше примеров обучения. Для чего это полезно, это помогает исправить высокую дисперсию.

0:45

И конкретно, если у вас есть проблема с высоким уровнем смещения и у вас нет проблемы с отклонениями, мы увидели в предыдущем видео, что больше примеров обучения,

0:54

в то время как, возможно, просто не поможет много. Таким образом, первый вариант полезен, только если вы, скажем, нарисуете кривые обучения и выясните, что у вас есть хотя бы небольшая вариация, а это означает, что ошибка перекрестной проверки, вы знаете, немного больше, чем ваш набор упражнений ошибка. Как насчет того, чтобы попробовать меньший набор функций? Ну, попробовав меньший набор функций, это опять-таки то, что фиксирует высокую дисперсию.

1:17

Иными словами, если вы выясните, просмотрев кривые обучения или что-то еще, что вы использовали, которые имеют высокую проблему смещения; то для добра, не тратьте свое время, пытайтесь тщательно выбрать меньший набор функций для использования. Потому что, если у вас проблема с высоким уровнем смещения, использование меньшего количества функций не поможет. В то время как, наоборот, если вы посмотрите на кривые обучения или что-то еще, вы выясните, что у вас проблема с большими отклонениями, то, действительно, пытайтесь выбрать меньший набор функций, это действительно может очень хорошо использовать ваше время. Как насчет того, чтобы пытаться получить дополнительные функции, добавляя функции, как правило, не всегда, но обычно мы думаем об этом как о решении

1:54

для устранения проблем с высоким уровнем смещения. Поэтому, если вы добавляете дополнительные функции, это обычно происходит потому, что

2:01

ваша текущая гипотеза слишком проста, и поэтому мы хотим попытаться получить дополнительные функции, чтобы наша гипотеза лучше соответствовала учебному набору. И аналогично, добавление полиномиальных признаков; это еще один способ добавления функций, и есть другой способ попытаться устранить проблему с высоким уровнем смещения.

2:21

И если конкретно, если ваши кривые обучения показывают вам, что у вас все еще есть проблема с большими отклонениями, то, знаете, снова это, возможно, менее эффективно использует ваше время.

2:30

И, наконец, уменьшение и увеличение λ . Это быстро и легко попробовать, я думаю, что эти менее вероятны, как известно, много лет вашей жизни. Но, уменьшая λ , вы уже знаете, что исправляет высокий уклон.

2:45

Если вам это непонятно, вы знаете, я рекомендую вам приостановить видео и продумать это, чтобы убедить себя, что уменьшение λ помогает исправить высокий уклон, в то время как увеличение λ сильно изменяет.

2:59

И если вы не знаете, почему это так, сделайте паузу в видео и убедитесь, что вы можете убедить себя, что это так. Или взгляните на кривые, которые мы рисовали в конце предыдущего видео, и постараемся понять, почему это так.

3:15

Наконец, давайте возьмем все, что мы узнали, и свяжем его с нейронными сетями, и вот, вот несколько практических советов, как я обычно выбираю архитектуру или схему подключения нейронных сетей, которые я использую.

3:30

Итак, если вы подстраиваетесь к нейронной сети, один из вариантов будет соответствовать, скажем, довольно маленькой нейронной сети, вы знаете, относительно мало скрытых единиц, может быть, только одна скрытая единица. Если вы настроите нейронную сеть, одним из вариантов будет установка относительно небольшой нейронной сети, например,

3:48

относительно немного, может быть, только один скрытый слой и, возможно, лишь относительно небольшое количество скрытых единиц. Таким образом, сеть, подобная этой, может иметь относительно немного параметров и быть более склонной к недоподготовке.

4:00

Основным преимуществом этих небольших нейронных сетей является то, что вычисления будут дешевле.

4:05

Альтернативой может быть установка, возможно, относительно большой нейронной сети с более скрытыми единицами - в ней есть много скрытых или более скрытых слоев.

4:16

И поэтому эти нейронные сети имеют больше параметров и, следовательно, более подвержены переобучению.

4:22

Один из недостатков, часто не главный, а что-то, о чем можно подумать, заключается в том, что если у вас в вашей сети большое количество нейронов, то это может быть более дорогостоящим.

4:33

Хотя, по разному, это, как мы надеемся, не является большой проблемой.

4:36

Основная потенциальная проблема этих гораздо более крупных нейронных сетей заключается в том, что она может быть более подвержена переобучению, и получается, что если вы применяете нейронную сеть очень часто, используя большую нейронную сеть, часто она на самом деле больше, тем лучше

4:50

но если это переобучение, вы можете использовать регуляризацию для перенаправления, обычно используя более крупную нейронную сеть, используя регуляризацию для адреса, перерабатывает, что часто более эффективно, чем использование меньшей нейронной сети. И главным возможным недостатком является то, что он может быть более дорогостоящим.

5:10

И, наконец, одним из других решений является, скажем, количество скрытых слоев, которые вы хотите иметь, не так ли? Итак, вам нужен один скрытый слой или вы хотите, чтобы три скрытых слоя, как мы показали здесь, или вам нужны два скрытых слоя?

5:23

И обычно, как я уже сказал в предыдущем видео, использование одного скрытого слоя является разумным дефолтом, но если вы хотите выбрать количество скрытых слоев, еще одна вещь, которую вы можете попробовать, это найти перекрестную проверку обучения, и раскол тестового набора и попробуйте обучить нейронные сети одним скрытым слоем или двумя скрытыми слоями или тремя скрытыми слоями и посмотреть, какая из этих нейронных сетей лучше всего работает на наборах перекрестной проверки. Вы берете три нейронные сети с одним, двумя и тремя скрытыми слоями и вычисляете ошибку перекрестной проверки в Jcv и все из них и используете это, чтобы выбрать, какая из них вы считаете лучшей нейронной сетью.

6:02

Таким образом, это касается смещения и дисперсии, а также способов обучения кривых, которые пытались диагностировать эти проблемы. Насколько вы думаете, подразумевается, для одного может быть правдивым или не правдивым, чтобы попытаться улучшить производительность алгоритма обучения.

6:16

Если вы поняли содержание последних нескольких видеороликов, и если вы их примените, вы на самом деле будете намного эффективнее и получите алгоритмы обучения для решения проблем и даже большую часть, может быть, большинство практикующих машинного обучения здесь, в Силиконовой долине, сегодня делают эти вещи, как их полный рабочий день.

6:35

Поэтому я надеюсь, что эти советы по опыту в диагностике

6:42

поможет вам эффективно и эффективно применять обучение и заставить их работать очень хорошо.