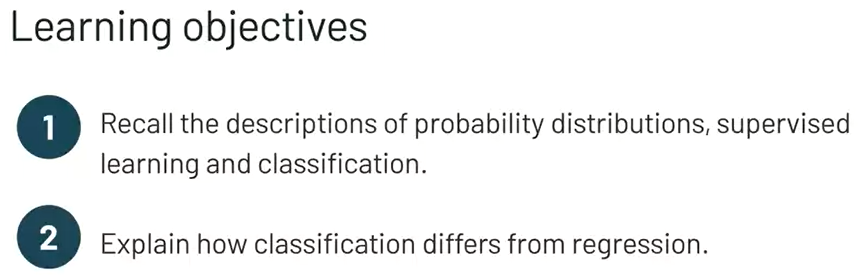
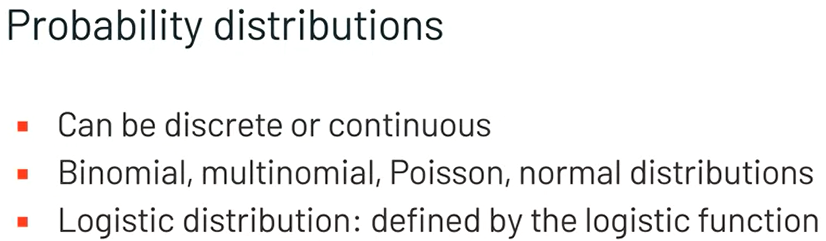
Classification Modelling

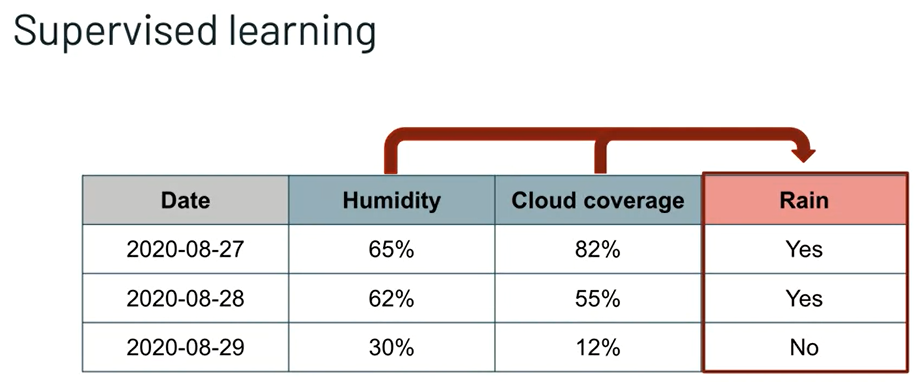
A Review of Key Concepts



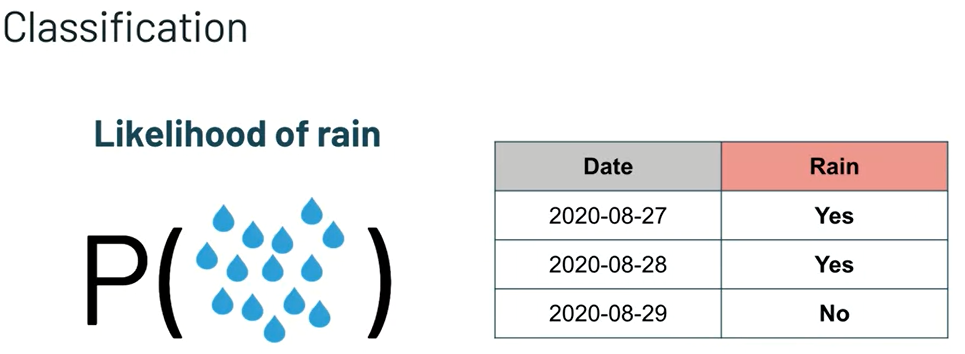
Welcome to Lesson 3, Classification Modeling. [Ders 3, Sınıflandırma Modelleme'ye hoş geldiniz.] In this lesson, you'll learn how to use and evaluate logistic regression for classification problems. [Bu derste, sınıflandırma problemleri için lojistik regresyonu nasıl kullanacağınızı ve değerlendireceğinizi öğreneceksiniz.] You will also learn some of the key concepts associated with classification modeling. [Ayrıca sınıflandırma modellemesi ile ilgili bazı temel kavramları da öğreneceksiniz.]



Previously in this course, we learned about some of the different kinds of probability distributions, for both discrete and continuous variables. [Bu kursun önceki bölümlerinde, hem kesikli hem de sürekli değişkenler için farklı olasılık dağılımlarından bazılarını öğrenmiştik.] Some of these were binomial, multinomial, Poisson, Bernoulli, and normal distributions. [Bunlardan bazıları binom, çok terimli, Poisson, Bernoulli ve normal dağılımlardı.] In addition, we also talked about a logistic distribution. [Ayrıca lojistik dağıtımdan da bahsettik.] The shape of this distribution is defined by the logistic function, it looks similar to the normal distribution, in that it's symmetric and has one peak, but it has a longer tail on one side. [Bu dağılımın şekli lojistik fonksiyon tarafından tanımlanır, simetrik olması ve bir tepe noktasına sahip olması, ancak bir tarafında daha uzun bir kuyruğu olması bakımından normal dağılıma benzer.] It's frequently used to model growth and for things like sports, physical sciences, finance, and other topics. [Büyümeyi modellemek ve spor, fizik bilimleri, finans ve diğer konular gibi şeyler için sıklıkla kullanılır.] It's a better fit than the normal distribution for extreme events or outliers. [Aşırı olaylar veya aykırı değerler için normal dağılımdan daha uygundur.]



We also learned previously about the distinction between supervised and unsupervised machine learning problems. [Denetimli ve denetimsiz makine öğrenimi sorunları arasındaki farkı da daha önce öğrenmiştik.] Supervised learning is when an algorithm is learning a function that maps a collection of inputs to an output based on example pairs of inputs and output, which is our training data. [Denetimli öğrenme, bir algoritmanın, eğitim verilerimiz olan örnek girdi ve çıktı çiftlerine dayalı olarak bir girdi koleksiyonunu bir çıktıya eşleyen bir işlevi öğrenmesidir.] In a way, our algorithm is being supervised by certain truth values for the function's output. [Bir bakıma algoritmamız, fonksiyonun çıktısı için belirli doğruluk değerleri tarafından denetleniyor.] These truth values, also called labels, are the key behind supervised learning. [Etiketler olarak da adlandırılan bu doğruluk değerleri, denetimli öğrenmenin ardındaki anahtardır.] We can't do supervised learning without these supervising labels. [Bu denetleyici etiketler olmadan denetimli öğrenme yapamayız.] This is in contrast to unsupervised learning, where algorithms look to understand relationships between inputs without any preexisting labels. [Bu, algoritmaların önceden var olan herhangi bir etiket olmadan girdiler arasındaki ilişkileri anlamaya çalıştığı denetimsiz öğrenmenin aksine.] In this module, we'll be focusing on classification, which is the type of machine learning tasks, within supervised learning. [Bu modülde, denetimli öğrenme kapsamında makine öğrenimi görevlerinin türü olan sınıflandırmaya odaklanacağız.]

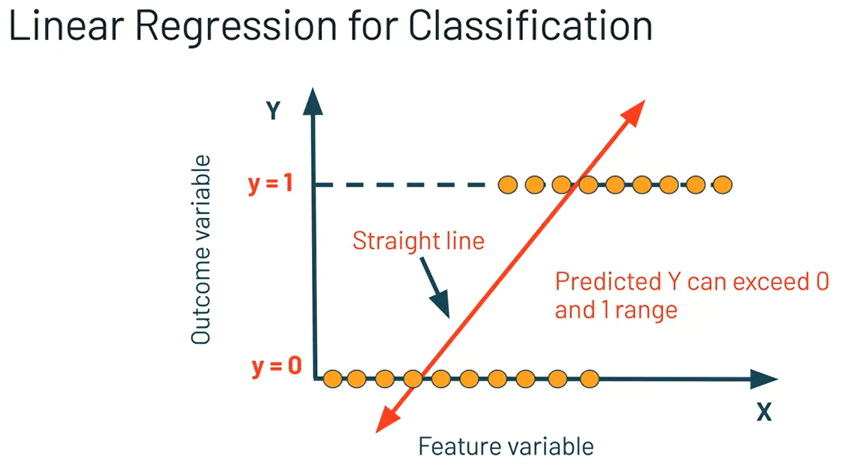


We previously defined classification as a subset of supervised learning, where the model is trying to predict which class out of a pre-defined set of classes a record belongs to. [Sınıflandırmayı daha önce, modelin bir kaydın önceden tanımlanmış bir sınıflar kümesinden hangi sınıfa ait olduğunu tahmin etmeye çalıştığı, denetimli öğrenmenin bir alt kümesi olarak tanımlamıştık.] Now that we've reviewed the descriptions of probability distributions, supervised learning, and classification, we'll move on to the next video, where we'll talk about logistic regression used as a classification modeling algorithm. [Olasılık dağılımlarının, denetimli öğrenmenin ve sınıflandırmanın açıklamalarını gözden geçirdiğimize göre, sınıflandırma modelleme algoritması olarak kullanılan lojistik regresyon hakkında konuşacağımız sonraki videoya geçeceğiz.]

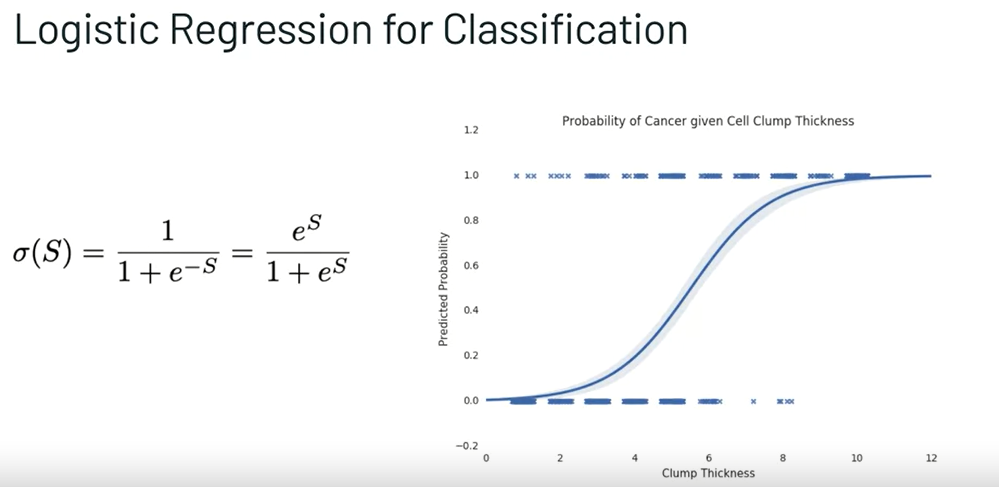
# An Introduction to Logistic Regression

# 

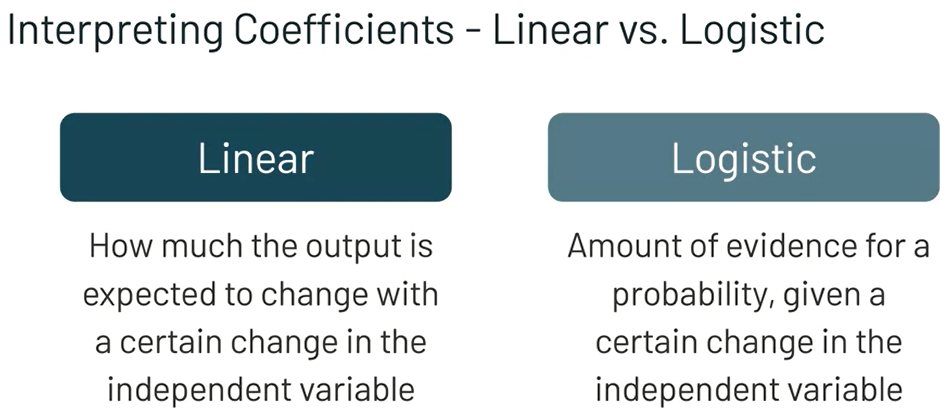
Within the field of machine learning we have several types of algorithms available to perform classification tasks. [Makine öğrenimi alanında, sınıflandırma görevlerini gerçekleştirmek için kullanılabilecek çeşitli algoritma türlerine sahibiz.]



One of the most commonly used is called logistic regression, which is what will be learning about in this video. [En yaygın olarak kullanılanlardan biri, bu videoda öğrenilecek olan lojistik regresyondur.] Previously we talked about some typical uses for Linear Regression, however, it's not well suited for classification tasks. [Daha önce Lineer Regresyonun bazı tipik kullanımlarından bahsetmiştik, ancak sınıflandırma görevleri için pek uygun değil.] This is because Linear Regression predicts continuous, not probabilistic outputs. [Bunun nedeni, Doğrusal Regresyon'un olası çıktıları değil, sürekliliği öngörmesidir.] In a binary classification problem, what we're interested in is the probability of an outcome occurring. [İkili bir sınıflandırma probleminde ilgilendiğimiz şey, bir sonucun meydana gelme olasılığıdır.] Probability is ranged between 0 and 1, where the probability of something that's definitely certain to happen is 1 and something unlikely to happen or definitely won't happen is 0. [Olasılık 0 ile 1 arasında değişir, burada kesinlikle olması kesin olan bir şeyin olasılığı 1 ve olması muhtemel olmayan veya kesinlikle olmayacak bir şeyin olasılığı 0'dır.] But in Linear Regression, were predicting an absolute number which can range outside 0 and 1 up to potentially infinity. [Ancak Lineer Regresyonda, 0 ve 1'in dışında potansiyel olarak sonsuzluğa kadar değişebilen mutlak bir sayı tahmin ediyordu.] If we try to draw a straight line to fit a function to the data, we see that it doesn't fit this binary probabilistic outcome very well at all. [Verilere bir fonksiyon sığdırmak için düz bir çizgi çizmeye çalışırsak, bu ikili olasılık sonucuna pek de uymadığını görürüz.] The function can predict a negative probability or a probability greater than 1, which would be a less than 0 or more than 100% chance of a given outcome. [İşlev, belirli bir sonucun 0'dan küçük veya %100'den fazla şansı olan negatif bir olasılığı veya 1'den büyük bir olasılığı tahmin edebilir.] That's obviously not realistic and doesn't make any sense. [Bu açıkça gerçekçi değil ve hiçbir anlam ifade etmiyor.] Another reason that Linear Regression doesn't perform classification well is that it doesn't work well with a very unbalanced data set. [Doğrusal Regresyon'un sınıflandırmayı iyi yapmamasının bir başka nedeni de, çok dengesiz bir veri seti ile iyi çalışmamasıdır.] Where there's much more of one class represented in the training data than another class. [Eğitim verilerinde başka bir sınıftan çok daha fazla bir sınıfın temsil edildiği yer.]

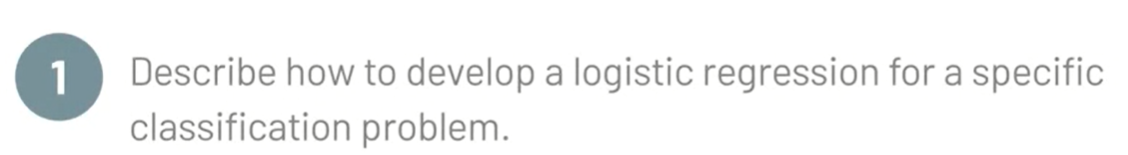


The Logistic function, which is what Logistic Regression uses, draws an S or a sigmoid curve to fit the data and will predict a probability between 0 and 1 as an output of the model. [Lojistik Regresyonun kullandığı Lojistik işlevi, verilere uyması için bir S veya sigmoid eğrisi çizer ve modelin çıktısı olarak 0 ile 1 arasında bir olasılık tahmin eder.] Here we see an example which is from a famous data set about Cancer and tumors. [Burada Kanser ve tümörler hakkında ünlü bir veri setinden bir örnek görüyoruz.] Where they probability of Cancer we see on the y-axis between 0 and 1, an Clump Thickness is one particular variable in the data set. [Y ekseninde 0 ile 1 arasında gördüğümüz Yengeç olasılığının olduğu yerde, bir Küme Kalınlığı veri setindeki belirli bir değişkendir.] And the equation on the left is just the sigmoid equation. [Ve soldaki denklem sadece sigmoid denklemdir.]



Interpreting the Coefficients of Logistic regression is trickier than it is for Linear regression. [Lojistik regresyonun Katsayılarını yorumlamak, Lineer regresyon için olduğundan daha zordur.] In Linear regression, the coefficients tell us if variable x goes up by a certain amount, how would we expect the value of the output to change? [Doğrusal regresyonda, katsayılar bize x değişkeninin belirli bir miktarda artması durumunda çıktının değerinin nasıl değişmesini bekleyeceğimizi söyler?] In Logistic regression, we can interpret a coefficient as the amount of evidence provided per change in the associated predictor variable or x, rather than a set amount. [Lojistik regresyonda, bir katsayıyı, belirli bir miktar yerine ilişkili tahmin değişkeni veya x'teki değişiklik başına sağlanan kanıt miktarı olarak yorumlayabiliriz.] We also say that the coefficient is the log odds of a particular outcome. [Katsayının belirli bir sonucun log oranları olduğunu da söylüyoruz.] It's important to also note that this coefficient is not constant across all values of the feature variable. [Bu katsayının, özellik değişkeninin tüm değerlerinde sabit olmadığına da dikkat etmek önemlidir.] Now that we've introduced some of the concepts of Logistic regression and how it differs from Linear regression in the next video, will talk about how to actually apply Logistic regression for a particular classification problem. [Şimdi, bir sonraki videoda Lojistik regresyonun bazı kavramlarını ve Lineer regresyondan nasıl farklı olduğunu tanıttığımıza göre, belirli bir sınıflandırma problemi için Lojistik regresyonun gerçekte nasıl uygulanacağı hakkında konuşacağız.]

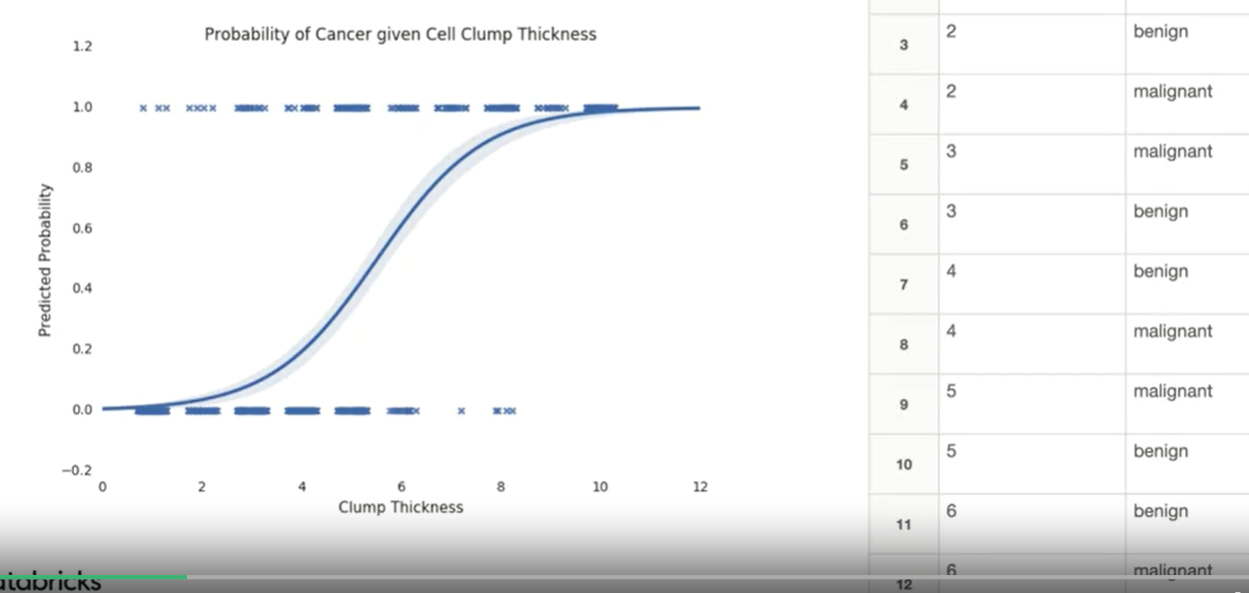
# Applying Logistic Regression



In this video, we'll walk-through an example of how to apply logistic regression to a particular problem. [Bu videoda, lojistik regresyonun belirli bir probleme nasıl uygulanacağına dair bir örnek üzerinden geçeceğiz.]



We'll do this using the Scikit-learn library that we previously learned a little bit about and we'll import the logistic regression model and create a logistic regression object, as you can see in the code block on the slide. [Bunu daha önce biraz öğrendiğimiz Scikit-learn kütüphanesini kullanarak yapacağız ve slayttaki kod bloğunda görebileceğiniz gibi lojistik regresyon modelini import edip bir lojistik regresyon nesnesi oluşturacağız.] Next, we'll identify our input and our output values from our dataset. [Ardından, veri kümemizden girdi ve çıktı değerlerimizi belirleyeceğiz.]

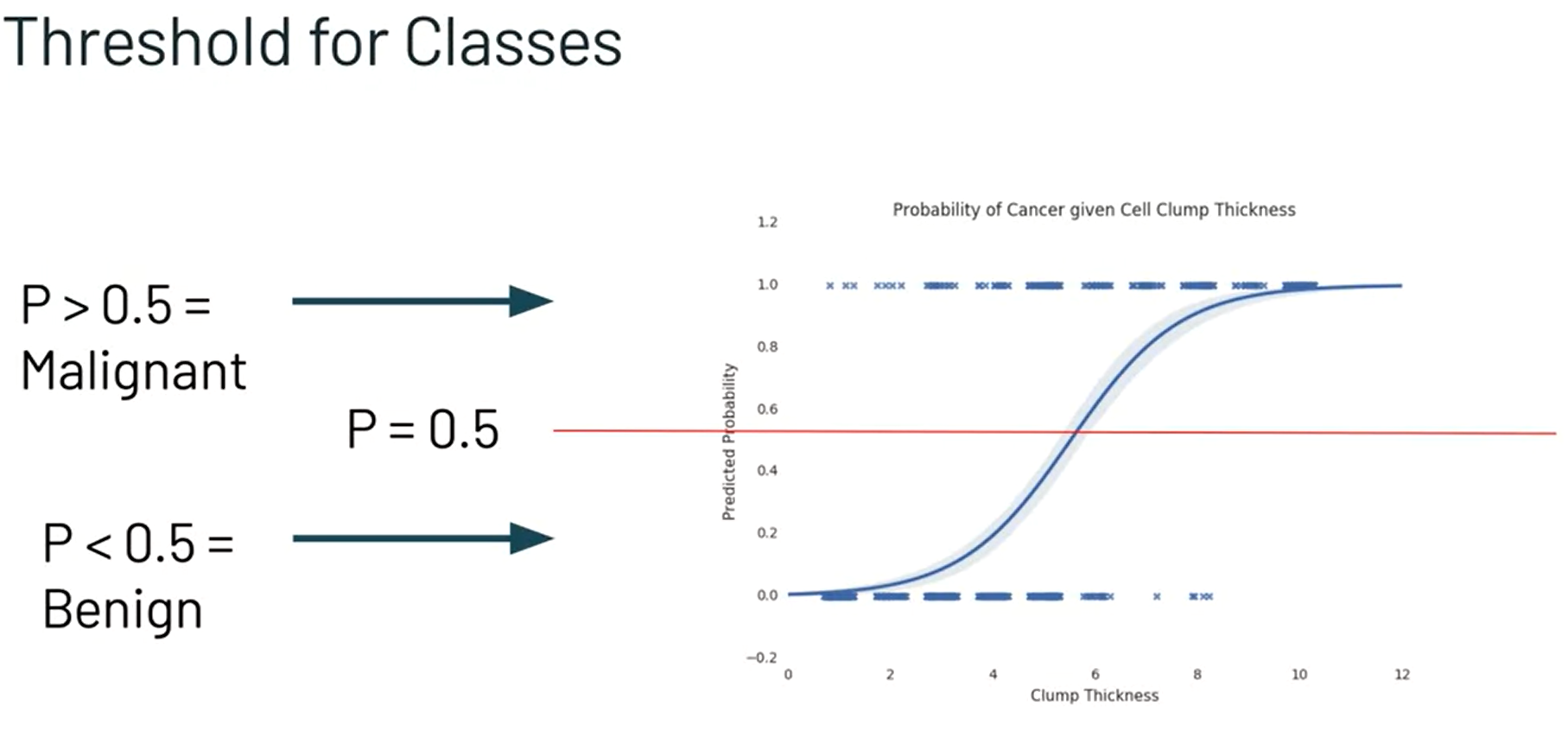


In this dataset our input is the variable clump thickness, which is a numeric variable. [Bu veri setinde girdimiz, sayısal bir değişken olan değişken küme kalınlığıdır.] The output is one of two classes. [Çıktı iki sınıftan biridir.] It has a class label of either benign or malignant. [İyi huylu veya kötü huylu bir sınıf etiketine sahiptir.] These class labels will get coded as zero or one for the sake of our Machine Learning model. [Bu sınıf etiketleri, Machine Learning modelimiz için sıfır veya bir olarak kodlanacaktır.] You can see in the graph on the slide, a reminder of what our logistic function will look like once we train our model on the clump thickness and class data. [Modelimizi küme kalınlığı ve sınıf verileri üzerinde eğittiğimizde lojistik fonksiyonumuzun nasıl görüneceğini hatırlatan slayttaki grafikte görebilirsiniz.] Our model will give us a predicted probability for each sample of whether it's benign or malignant. [Modelimiz, iyi huylu veya kötü huylu olup olmadığına dair her örnek için bize tahmin edilen bir olasılık verecektir.] Next we'll train our model on the training set by fitting it to the x train and y train values and then we can view the coefficients for the model. [Daha sonra modelimizi x dizisi ve y dizisi değerlerine uydurarak eğitim seti üzerinde eğiteceğiz ve ardından modelin katsayılarını örebiliriz.] In this case we have the beta-zero or intercept coefficient. [Bu durumda beta-sıfır veya kesişim katsayısına sahibiz.] Then the beta one, which is the weight that the model gives to the clump thickness coefficient. [Ardından, modelin küme kalınlık katsayısına verdiği ağırlık olan beta bir.] As we can see, it has a pretty high weight ecause in our sample model it is the only variable. [Gördüğümüz gibi oldukça yüksek bir ağırlığa sahip çünkü örnek modelimizde tek değişken o.] That was just a quick walk-through of how you would do a very simple logistic regression model with a dataset that has one input variable and predicts one of two classes for the output variable. [Bu, bir girdi değişkeni olan ve çıktı değişkeni için iki sınıftan birini öngören bir veri kümesiyle çok basit bir lojistik regresyon modelini nasıl yapacağınıza dair hızlı bir gözden geçirmeydi.]

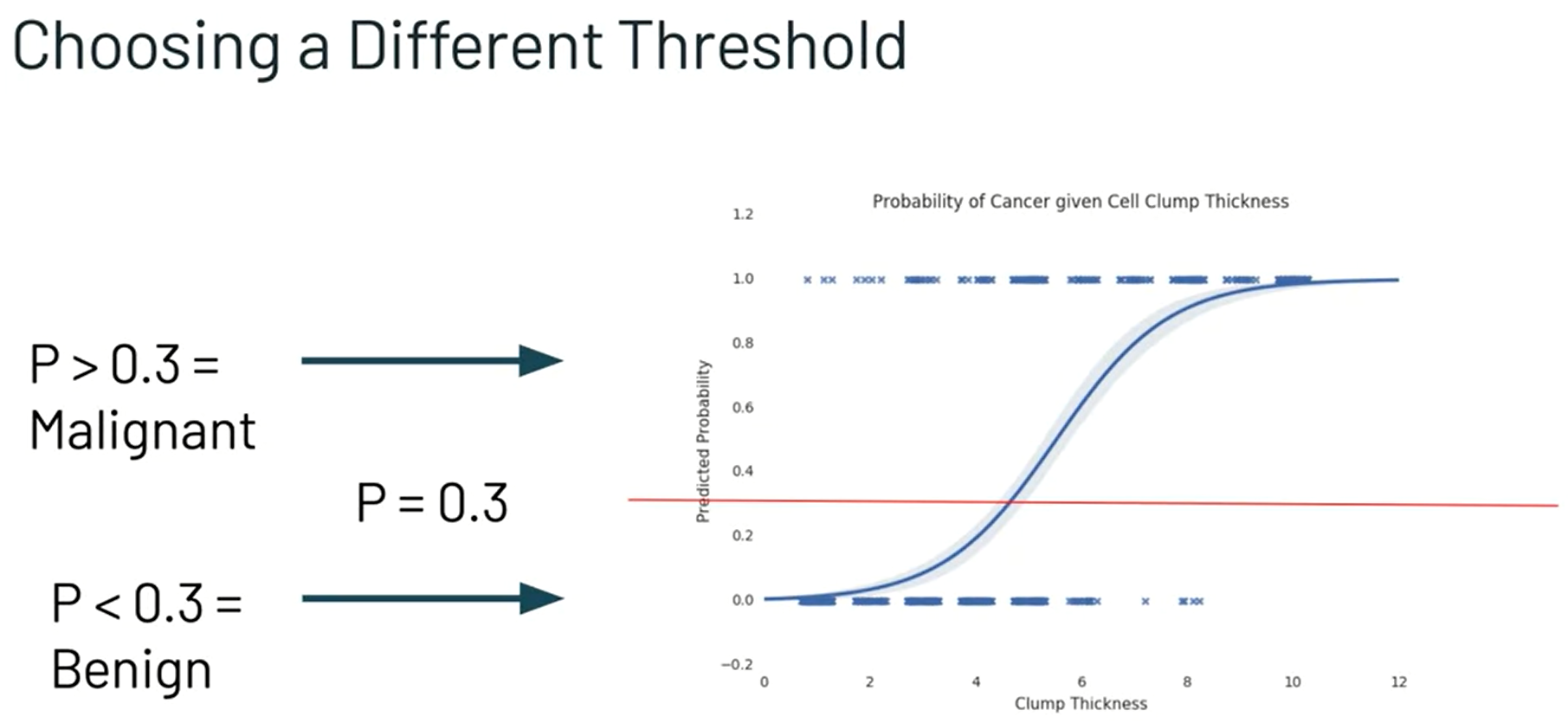
# Assigning Classes Based on Probabilities

# 

In this video, we'll look at how we assign label classes based on the predicted probabilities from a logistic regression model. [Bu videoda, bir lojistik regresyon modelinden tahmin edilen olasılıklara dayalı olarak etiket sınıflarını nasıl atadığımıza bakacağız.] In the previous video, we watched through a very simple example of training a logistic regression model. [Bir önceki videoda çok basit bir lojistik regresyon modeli eğitimi örneğini izledik.] We saw that we got a coefficient, but the model would also give us predicted label classes, and those are based on probabilities. [Bir katsayı elde ettiğimizi gördük, ancak model bize tahmin edilen etiket sınıflarını da verecekti ve bunlar olasılıklara dayanıyor.]



Since the logistic regression model returns probabilities of an observation being of a given class, which means it doesn't actually assign a class to each data point. [Lojistik regresyon modeli, belirli bir sınıfa ait bir gözlemin olasılıklarını döndürdüğünden, bu, aslında her veri noktasına bir sınıf atamadığı anlamına gelir.] We, as the data scientists, need to decide what cut-off threshold between 0-1 that we'll use to classify a sample as belonging to either class. [Veri bilimcileri olarak, bir örneği her iki sınıfa ait olarak sınıflandırmak için kullanacağımız 0-1 arasında hangi kesme eşiğine karar vermemiz gerekiyor.] This means that we identify a probability level where we decide whether, in this case, the sample is benign or malignant. [Bu, bu durumda örneğin iyi huylu veya kötü huylu olup olmadığına karar verdiğimiz bir olasılık seviyesi belirlediğimiz anlamına gelir.] The default threshold value is 0.5. [Varsayılan eşik değeri 0,5'tir.] We split the probabilities evenly between the two classes. [Olasılıkları iki sınıf arasında eşit olarak böldük.] But there are different reasons why. [Ama bunun farklı nedenleri var.] Again, as a data scientists using your knowledge and intuition, you might want or need to change this threshold value. [Yine, bilginizi ve sezginizi kullanan bir veri bilimci olarak, bu eşik değerini değiştirmek isteyebilirsiniz veya buna ihtiyacınız olabilir.]



Here, we see an example of what it would look like if we lower our threshold to 0.3 to predict malignant. [Burada, maligniteyi tahmin etmek için eşiğimizi 0,3'e düşürürsek nasıl görüneceğine dair bir örnek görüyoruz.] If we set our threshold lower, it means that for any predicted probability higher or equal to 0.3, we would say that sample belongs to the main malignant class. [Eşiğimizi daha düşük ayarlarsak, bu, 0,3'e eşit veya daha yüksek herhangi bir tahmin edilen olasılık için, örneğin ana malign sınıfa ait olduğunu söyleyebiliriz.] Whereas, if it's less than 0.3, we would classify it as benign. [Oysa 0,3'ten küçükse, onu iyi huylu olarak sınıflandırırız.] You're probably wondering why we would do this. [Muhtemelen bunu neden yaptığımızı merak ediyorsundur.] We would change this threshold depending on whether we're more concerned about false positives or false negatives. [Bu eşiği, yanlış pozitifler veya yanlış negatifler hakkında daha fazla endişe duymamıza bağlı olarak değiştirirdik.] Obviously, if we're lowering the threshold to classify a sample as malignant, it means that we're trying harder to make sure we don't miss any of the potentially malignant samples. [Açıkçası, bir numuneyi kötü huylu olarak sınıflandırma eşiğini düşürüyorsak, bu, potansiyel olarak kötü huylu örneklerin hiçbirini kaçırmamak için daha çok çalıştığımız anlamına gelir.] At the expense of over-classifying samples as malignant rather than benign. [Numunelerin iyi huylu olmaktan çok kötü huylu olarak sınıflandırılması pahasına.] Maybe in this example, a particular type of tumor is really deadly. [Belki bu örnekte, belirli bir tümör türü gerçekten ölümcüldür.] In this case, we would want to error on the side of caution to have more of these false positives than the false negatives. [Bu durumda, yanlış negatiflerden daha fazla bu yanlış pozitiflere sahip olmak için dikkatli olmak isteriz.] Let's say you want to be really careful to identify all possible cases where it's malignant. [Kötü huylu olduğu tüm olası vakaları belirlemek için gerçekten dikkatli olmak istediğinizi varsayalım.] We're airing on the side of over-predicting that. [Bunu fazla tahmin etme tarafında yayın yapıyoruz.] That's why we lower the threshold. [Bu yüzden eşiği düşürüyoruz.] This means that the model will be more sensitive to predict a malignant tumor. [Bu, modelin kötü huylu bir tümörü tahmin etmede daha duyarlı olacağı anlamına gelir.] On the other hand, if the type of tumor was less deadly and maybe we were more concerned with not wanting to cause undue worry or concern or leading to unnecessary extra testing or biopsies, again, if this particular type of tumor was fairly deadly, we would err on the side of having more false negatives. [Öte yandan, tümörün türü daha az ölümcül olsaydı ve belki de gereksiz endişe veya endişeye neden olmamak veya gereksiz ekstra testlere veya biyopsilere yol açmamak konusunda daha fazla endişe duyuyorsak, yine, bu belirli tümör türü oldukça ölümcül olsaydı, daha fazla yanlış negatife sahip olma tarafında hata olur.] Therefore, we would want our model to be more specific rather than sensitive and we could raise the threshold. [Bu nedenle modelimizin hassas değil daha spesifik olmasını isterdik ve eşiği yükseltebiliriz.] If we weren't that concerned about malignant tumors in this case, then we could raise the threshold. [Bu vakada kötü huylu tümörler konusunda o kadar endişeli olmasaydık, eşiği yükseltebilirdik.] Maybe we raise it to 0.7. [Belki 0,7'ye yükseltiriz.] That's our threshold for saying whether a sample is malignant or benign. [Bir örneğin kötü huylu mu yoksa iyi huylu mu olduğunu söylemek için eşiğimiz budur.] Now that we've learned about choosing different thresholds based on what our goal for our logistic regression model is, in the next videos, we'll learn about evaluating and interpreting the output of the logistic regression model. [Lojistik regresyon modelimiz için hedefimizin ne olduğuna bağlı olarak farklı eşikler seçmeyi öğrendiğimize göre, sonraki videolarda lojistik regresyon modelinin çıktısını değerlendirmeyi ve yorumlamayı öğreneceğiz.]

# Classification Evaluation

Now that we've seen how to build a simple logistic regression model for classification. [Artık sınıflandırma için basit bir lojistik regresyon modelinin nasıl oluşturulacağını gördük.] Seeing how we would train it on some data and tested it on new data. [Onu bazı veriler üzerinde nasıl eğiteceğimizi ve yeni veriler üzerinde test edeceğimizi görmek.] We now have to evaluate how well our model performed and we will do this using classification evaluation metrics. [Şimdi modelimizin ne kadar iyi performans gösterdiğini değerlendirmemiz gerekiyor ve bunu sınıflandırma değerlendirme metriklerini kullanarak yapacağız.] On this side we see an example output that compares the predicted classes versus the actual classes. [Bu tarafta, tahmin edilen sınıfları gerçek sınıflarla karşılaştıran bir örnek çıktı görüyoruz.] So the middle column is malignant is whether that sample was actually malignant in our data set, whereas the third column prediction is what our model decided and again remember that this is the 01. [Yani orta sütun kötü huyludur, örneğin veri setimizde gerçekten kötü huylu olup olmadığıdır, oysa üçüncü sütun tahmini modelimizin karar verdiği şeydir ve bunun 01 olduğunu tekrar hatırlayın.] The class prediction is based entirely on our threshold. [Sınıf tahmini tamamen bizim eşiğimize dayanmaktadır.] So the model predicts probabilities but then it filters that through our threshold level. [Yani model olasılıkları tahmin ediyor ama sonra bunu bizim eşik seviyemize göre filtreliyor.] So if the probability if it's the default threshold of 0.5, if the model determined that sample one in the first row had a probability of 0.2 for being malignant. [Yani, olasılık varsayılan eşik 0,5 ise, model ilk satırdaki örneğin kötü huylu olma olasılığının 0,2 olduğunu belirlediyse.] It would say that it belongs to class zero which means benign. [İyi huylu anlamına gelen sıfır sınıfına ait olduğunu söylerdi.] So we can see that overall it looks like the model probably did fairly well. [Dolayısıyla, genel olarak modelin muhtemelen oldukça iyi iş çıkarmış gibi göründüğünü görebiliriz.] Many of the predictions and actual values line up but we also see some mistakes. [Tahminlerin ve gerçek değerlerin çoğu aynı hizada ama bazı hatalar da görüyoruz.] So for example in row four the actual sample was benign but the model predicted that it was malignant. [Örneğin dördüncü satırda gerçek örnek iyi huyluydu ama model bunun kötü huylu olduğunu öngördü.] And then we also see on row 13 the opposite where the actual sample was malignant but the model predicted that it was not. [Ve ayrıca 13. satırda, gerçek örneğin kötü huylu olduğu, ancak modelin öyle olmadığını öngördüğü yerin tam tersini görüyoruz.] From looking at the predicted versus actual values we can find out the accuracy of our model. [Tahmin edilen ve gerçek değerlere bakarak modelimizin doğruluğunu öğrenebiliriz.] So simple accuracy equation is the number of true positives plus the number of true negatives that the model predicted divided by the true positives plus the true negatives plus the false positives plus the false negatives. [Çok basit doğruluk denklemi, gerçek pozitiflerin sayısı artı modelin öngördüğü gerçek negatiflerin sayısı bölü gerçek pozitifler artı gerçek negatifler artı yanlış pozitifler artı yanlış negatifler.] So basically this boils down to, we're looking at the number of predictions the model got correct, divided by the total number of predictions that it made. [Yani temelde bu, modelin doğru yaptığı tahminlerin sayısının, yaptığı toplam tahmin sayısına bölünmesine bakarız.] We can also break this down by looking at a confusion matrix which is a table that reports the number of false positives, false negatives, true positives and true negatives. [Bunu, yanlış pozitiflerin, yanlış negatiflerin, gerçek pozitiflerin ve gerçek negatiflerin sayısını bildiren bir tablo olan bir karışıklık matrisine bakarak da çözebiliriz.] This confusion matrix is the basis of a number of different evaluation metrics like we saw with accuracy but also for precision or something called the area under the ROC curve, which we won't go into the details here but it's just good to mention so you're familiar with the term. [Bu karışıklık matrisi, doğrulukla ve aynı zamanda kesinlik veya ROC eğrisinin altındaki alan adı verilen bir şey için gördüğümüz gibi bir dizi farklı değerlendirme metriğinin temelidir, burada ayrıntılara girmeyeceğiz ancak bahsetmekte fayda var. terimine aşinayız.] For instance, as we saw, accuracy is calculated by adding the true positives and the true negatives and dividing by the total number of records. [Örneğin, gördüğümüz gibi doğruluk, gerçek pozitifler ve gerçek negatifler toplanarak ve toplam kayıt sayısına bölünerek hesaplanır.] So based on the confusion matrix, we could see that there are 165 true positives and 437 true negatives. [Karışıklık matrisine dayanarak, 165 gerçek pozitif ve 437 gerçek negatif olduğunu görebiliriz.] And then we would divide the sum of those two by 699 which is the total number of predictions. [Ve sonra bu ikisinin toplamını toplam tahmin sayısı olan 699'a bölerdik.] We also see what's known as Type one and Type two errors. [Ayrıca Type bir ve Type iki hataları olarak bilinenleri de görüyoruz.] This is something you may have heard of before. [Bu daha önce duymuş olabileceğiniz bir şey.] A type one error is a false positive which if we translate that back into hypothesis testing false positive means that we have rejected a true null hypothesis. [Birinci tip hata, yanlış pozitiftir; bunu tekrar hipotez testine çevirirsek, yanlış pozitif, gerçek bir boş hipotezi reddettiğimiz anlamına gelir.] So again we rejected the null hypothesis incorrectly because the null hypothesis was true. [Yani sıfır hipotezi doğru olduğu için boş hipotezi yanlış bir şekilde reddettik.] And a type two error is a false negative where we failed to reject a null hypothesis that was actually false. [Ve ikinci tip hata, aslında yanlış olan bir sıfır hipotezini reddetmeyi başaramadığımız yanlış bir negatiftir.] This leads us into the concepts of precision vs. [Bu bizi hassasiyet ve hassasiyet kavramlarına götürür.] Recall, which are another way to look at evaluating how well our model did and this actually breaks down accuracy into two different components. [Modelimizin ne kadar iyi çalıştığını değerlendirmenin başka bir yolu olan geri çağırma, doğruluğu iki farklı bileşene ayırır.] So precision which is also known as specificity. [Yani özgüllük olarak da bilinen kesinlik.] And we can see the equation on the slide precision is just asking ourselves of the cases that the model predicted to be positive what percentage were actually positive. [Ve slayttaki denklemi görebiliriz ki, kendimize sadece modelin pozitif olarak tahmin ettiği durumlardan yüzde kaçının gerçekten pozitif olduğunu soruyoruz.] And recall asks of the cases that were actually positive. [Ve hatırlama, aslında olumlu olan vakaları soruyor.] What percentage did the model correctly identify as positive? [Model yüzde kaçı doğru olarak pozitif olarak tanımladı?] In our classification example precision would measure the proportion of us that we predicted to be malignant that were actually malignant. [Sınıflandırma örneğimizde kesinlik, kötü huylu olduğunu tahmin ettiğimiz ve aslında kötü huylu olan oranımızı ölçecektir.] And recall in our example would mean correctly identifying true positives. [Ve örneğimizdeki hatırlama, gerçek pozitifleri doğru bir şekilde tanımlamak anlamına gelir.] So for all of the tumor hat are actually malignant recall would tell us how many we correctly identified as being malignant. [Bu nedenle, tüm tümör şapkasının aslında malign olduğunu hatırlamak bize kaç tanesini doğru olarak malign olarak tanımladığımızı söyleyecektir.] Although we generally do want to aim for high precision and high recall achieving both at the same time isn't possible. [Genelde yüksek hassasiyet ve yüksek geri çağırmayı hedeflemek istesek de, her ikisini aynı anda elde etmek mümkün değildir.] For example if we change the model to one that gives us a high recall and again remember that this is based on changing our threshold value. [Örneğin, modeli bize yüksek hatırlama sağlayan bir modelle değiştirirsek ve bunun eşik değerimizi değiştirmeye dayandığını tekrar hatırlayın.] So if we change the model to a high recall we might detect all of the patients in this example who actually have heart disease, but we might end up giving treatments to a lot of patients who don't suffer from it. [Dolayısıyla, modeli yüksek hatırlamaya değiştirirsek, bu örnekte gerçekten kalp hastalığı olan tüm hastaları tespit edebiliriz, ancak sonuçta kalp hastalığı olmayan birçok hastaya tedavi verebiliriz.] Likewise, if we aim for high precision because we want to avoid giving any wrong or unrequired treatment, we end up getting a lot of patients who actually have a heart disease who don't receive any treatment. [Aynı şekilde, yanlış veya gereksiz herhangi bir tedavi vermekten kaçınmak istediğimiz için yüksek kesinliği hedeflersek, aslında kalp hastalığı olan ve herhangi bir tedavi almayan çok sayıda hasta elde ederiz.] So we need to find some kind of trade off between precision and recall. [Bu yüzden kesinlik ve geri çağırma arasında bir çeşit değiş tokuş bulmamız gerekiyor.] We first need to decide which of these is more important for our particular classification problem. [Öncelikle, belirli sınıflandırma problemimiz için bunlardan hangisinin daha önemli olduğuna karar vermemiz gerekiyor.] So would we want to accidentally over medicaid or would we want to inadvertently under medicaid? [Öyleyse, yanlışlıkla medicaid'i mi yoksa istemeden medicaid'i mi almak isteriz?] And it comes down to asking which would be more damaging or less desirable. [Ve hangisinin daha zararlı veya daha az arzu edilir olduğunu sormak geliyor.] In other examples, we might just make a different decision. [Diğer örneklerde, sadece farklı bir karar verebiliriz.] So if our classification model was trying to identify whether a bank customer would default on a loan or not, it would be more desirable to have a high precision since the bank wouldn't want to lose customers who were denied a loan based on the model's prediction that they would be defaulters, which would leave a lot of angry customers. [Dolayısıyla, sınıflandırma modelimiz bir banka müşterisinin bir krediyi temerrüde düşüp düşmeyeceğini belirlemeye çalışıyorsa, yüksek bir kesinliğe sahip olmak daha arzu edilir olacaktır, çünkü banka, modele dayalı olarak kredi verilmeyen müşterileri kaybetmek istemeyecektir. temerrüde düşeceklerine dair tahmin, bu da pek çok kızgın müşteri bırakacaktı.] There's also a lot of situations where precision and recall are actually equally important. [Ayrıca kesinlik ve hatırlamanın aslında eşit derecede önemli olduğu birçok durum vardır.] For example, in this model, where we're trying to identify the patients with heart disease. [Örneğin, kalp hastalığı olan hastaları tanımlamaya çalıştığımız bu modelde.] If the doctor informs us that the patients who were incorrectly classified as having heart disease are equally important to identify, since they could be indicative of some other ailments, then we would aim for not just a high recall, but a high precision as well. [Doktor, yanlış bir şekilde kalp hastalığı olarak sınıflandırılan hastaların, diğer bazı rahatsızlıkların göstergesi olabileceğinden, teşhis edilmesinin eşit derecede önemli olduğunu bize bildirirse, o zaman sadece yüksek bir hatırlama değil, aynı zamanda yüksek bir kesinlik de hedeflemiş oluruz.] In cases like this, we use something called the F1 score. [Bu gibi durumlarda F1 puanı denen bir şey kullanırız.] The F1 score is described as the harmonic mean of precision and recall where we're trying to find the perfect balance between them. [F1 puanı, hassaslığın harmonik ortalaması ve aralarındaki mükemmel dengeyi bulmaya çalıştığımız yeri hatırlama olarak tanımlanır.] So we could just aim for a good F1 score, and that would give us exactly what we're trying to achieve the balance between precision and recall. [Yani sadece iyi bir F1 puanı hedefleyebilirdik ve bu bize kesinlik ve hatırlama arasındaki dengeyi sağlamaya çalıştığımız şeyi verirdi.]