DSE6211 Final Project

Joseph Annand

2024-03-09

## Libraries

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(lubridate)

##   
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':  
##   
## date, intersect, setdiff, union

library(caret)

## Warning: package 'caret' was built under R version 4.3.2

## Loading required package: ggplot2

## Loading required package: lattice

library(MESS)

## Warning: package 'MESS' was built under R version 4.3.2

## Data Processing

project\_data <- read.csv("project\_data/project\_data.csv")  
  
rownames(project\_data) <- project\_data$Booking\_ID  
project\_data <- project\_data[, -1]  
  
training\_ind <- createDataPartition(project\_data$booking\_status,  
 p = 0.75,  
 list = F,  
 times = 1)  
  
training\_set <- project\_data[training\_ind, ]  
test\_set <- project\_data[-training\_ind, ]  
  
  
training\_set$booking\_status <- ifelse(training\_set$booking\_status=="canceled",1,0)  
test\_set$booking\_status <- ifelse(test\_set$booking\_status=="canceled",1,0)  
  
  
###### Training Set Categorical Features  
  
top\_8\_previous\_not\_cancelled <- training\_set %>%  
 group\_by(no\_of\_previous\_bookings\_not\_canceled) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 select(no\_of\_previous\_bookings\_not\_canceled) %>%  
 slice(1:8)  
  
top\_2\_number\_of\_children <- training\_set %>%  
 group\_by(no\_of\_children) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 slice(1:2)  
  
top\_3\_previous\_cancellations <- training\_set %>%  
 group\_by(no\_of\_previous\_cancellations) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 slice(1:3)  
  
top\_8\_week\_nights <- training\_set %>%  
 group\_by(no\_of\_week\_nights) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 slice(1:8)  
  
top\_6\_weekend\_nights <- training\_set %>%  
 group\_by(no\_of\_weekend\_nights) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 slice(1:6)  
  
top\_4\_special\_requests <- training\_set %>%  
 group\_by(no\_of\_special\_requests) %>%  
 summarise(count = n()) %>%  
 arrange(desc(count)) %>%  
 slice(1:4)  
  
  
training\_set$no\_of\_previous\_bookings\_not\_canceled <- ifelse(  
 training\_set$no\_of\_previous\_bookings\_not\_canceled %in% top\_8\_previous\_not\_cancelled$no\_of\_previous\_bookings\_not\_canceled,  
 training\_set$no\_of\_previous\_bookings\_not\_canceled, "8+"  
)  
  
training\_set$no\_of\_children <- ifelse(  
 training\_set$no\_of\_children %in% top\_2\_number\_of\_children$no\_of\_children,  
 training\_set$no\_of\_children, "3+"  
)  
  
training\_set$no\_of\_previous\_cancellations <- ifelse(  
 training\_set$no\_of\_previous\_cancellations %in% top\_3\_previous\_cancellations$no\_of\_previous\_cancellations,  
 training\_set$no\_of\_previous\_cancellations, "3+"  
)  
  
training\_set$no\_of\_week\_nights <- ifelse(  
 training\_set$no\_of\_week\_nights %in% top\_8\_week\_nights$no\_of\_week\_nights,  
 training\_set$no\_of\_week\_nights, "8+"  
)  
  
training\_set$no\_of\_weekend\_nights <- ifelse(  
 training\_set$no\_of\_weekend\_nights %in% top\_6\_weekend\_nights$no\_of\_weekend\_nights,  
 training\_set$no\_of\_weekend\_nights, "6+"  
)  
  
training\_set$type\_of\_meal\_plan <- ifelse(training\_set$type\_of\_meal\_plan %in% c("meal\_plan\_1", "meal\_plan\_2"),  
 training\_set$type\_of\_meal\_plan,  
 "other")  
  
training\_set$no\_of\_special\_requests <- ifelse(training\_set$no\_of\_special\_requests %in% top\_4\_special\_requests$no\_of\_special\_requests,  
 training\_set$no\_of\_special\_requests,  
 "4+")  
  
training\_set$arrival\_date <- parse\_date\_time(training\_set$arrival\_date, "ymd")  
training\_set$booking\_date <- int\_start(interval(training\_set$arrival\_date - ddays(training\_set$lead\_time),   
 training\_set$arrival\_date))  
  
season\_months <- data.frame(winter = c(12,1,2), spring = c(3,4,5),  
 summer = c(6,7,8), fall = c(9,10,11))  
  
  
get\_season <- function(x) {  
 y <- month(x)  
 # print(y)  
 for (j in 1:length(colnames(season\_months))) {  
 # print(j)  
 if (y %in% season\_months[[j]]) {  
 # print(colnames(season\_months)[j])  
 return(colnames(season\_months)[j])  
 }  
 }  
}  
  
training\_set$arrival\_season <- sapply(training\_set$arrival\_date, get\_season)  
training\_set$arrival\_day <- wday(training\_set$arrival\_date)  
training\_set$arrival\_day <- ifelse(training\_set$arrival\_day %in% c(1,6,7),  
 "weekend", "weekday")  
  
  
categorical\_var <- c(1:7,10:13,15,18,19)  
for (i in categorical\_var) {  
 training\_set[[i]] <- factor(training\_set[[i]])  
}  
  
cat\_col <- colnames(training\_set[, c(1:7,10:13,15,18,19)])  
  
onehot\_encoder <- dummyVars(~ no\_of\_adults + no\_of\_children + no\_of\_weekend\_nights  
 + no\_of\_week\_nights + type\_of\_meal\_plan + required\_car\_parking\_space  
 + room\_type\_reserved + market\_segment\_type + repeated\_guest  
 + no\_of\_previous\_cancellations + no\_of\_previous\_bookings\_not\_canceled  
 + no\_of\_special\_requests + arrival\_season + arrival\_day,  
 training\_set[, c("no\_of\_adults","no\_of\_children","no\_of\_weekend\_nights",  
 "no\_of\_week\_nights","type\_of\_meal\_plan",  
 "required\_car\_parking\_space",  
 "room\_type\_reserved","market\_segment\_type",  
 "repeated\_guest","no\_of\_previous\_cancellations",  
 "no\_of\_previous\_bookings\_not\_canceled",  
 "no\_of\_special\_requests","arrival\_season",  
 "arrival\_day")],  
 levelsOnly = F,  
 fullRank = T)  
  
onehot\_enc\_training <- predict(onehot\_encoder, training\_set[, c("no\_of\_adults","no\_of\_children","no\_of\_weekend\_nights",  
 "no\_of\_week\_nights","type\_of\_meal\_plan",  
 "required\_car\_parking\_space",  
 "room\_type\_reserved","market\_segment\_type",  
 "repeated\_guest","no\_of\_previous\_cancellations",  
 "no\_of\_previous\_bookings\_not\_canceled",  
 "no\_of\_special\_requests","arrival\_season",  
 "arrival\_day")])  
  
training\_set <- cbind(training\_set, onehot\_enc\_training)  
  
####### Test Set Categorical Variables  
  
test\_set$no\_of\_previous\_bookings\_not\_canceled <- ifelse(  
 test\_set$no\_of\_previous\_bookings\_not\_canceled %in% top\_8\_previous\_not\_cancelled$no\_of\_previous\_bookings\_not\_canceled,  
 test\_set$no\_of\_previous\_bookings\_not\_canceled, "8+"  
)  
  
test\_set$no\_of\_children <- ifelse(  
 test\_set$no\_of\_children %in% top\_2\_number\_of\_children$no\_of\_children,  
 test\_set$no\_of\_children, "3+"  
)  
  
test\_set$no\_of\_previous\_cancellations <- ifelse(  
 test\_set$no\_of\_previous\_cancellations %in% top\_3\_previous\_cancellations$no\_of\_previous\_cancellations,  
 test\_set$no\_of\_previous\_cancellations, "3+"  
)  
  
test\_set$no\_of\_week\_nights <- ifelse(  
 test\_set$no\_of\_week\_nights %in% top\_8\_week\_nights$no\_of\_week\_nights,  
 test\_set$no\_of\_week\_nights, "8+"  
)  
  
test\_set$no\_of\_weekend\_nights <- ifelse(  
 test\_set$no\_of\_weekend\_nights %in% top\_6\_weekend\_nights$no\_of\_weekend\_nights,  
 test\_set$no\_of\_weekend\_nights, "6+"  
)  
  
test\_set$type\_of\_meal\_plan <- ifelse(test\_set$type\_of\_meal\_plan %in% c("meal\_plan\_1", "meal\_plan\_2"),  
 test\_set$type\_of\_meal\_plan,  
 "other")  
  
test\_set$no\_of\_special\_requests <- ifelse(test\_set$no\_of\_special\_requests %in% top\_4\_special\_requests$no\_of\_special\_requests,  
 test\_set$no\_of\_special\_requests,  
 "4+")  
  
test\_set$arrival\_date <- parse\_date\_time(test\_set$arrival\_date, "ymd")  
test\_set$booking\_date <- int\_start(interval(test\_set$arrival\_date - ddays(test\_set$lead\_time),   
 test\_set$arrival\_date))  
  
  
test\_set$arrival\_season <- sapply(test\_set$arrival\_date, get\_season)  
test\_set$arrival\_day <- wday(test\_set$arrival\_date)  
test\_set$arrival\_day <- ifelse(test\_set$arrival\_day %in% c(1,6,7),  
 "weekend", "weekday")  
  
  
categorical\_var <- c(1:7,10:13,15,18,19)  
for (i in categorical\_var) {  
 test\_set[[i]] <- factor(test\_set[[i]])  
}  
  
cat\_col <- colnames(test\_set[, c(1:7,10:13,15,18,19)])  
  
onehot\_encoder <- dummyVars(~ no\_of\_adults + no\_of\_children + no\_of\_weekend\_nights  
 + no\_of\_week\_nights + type\_of\_meal\_plan + required\_car\_parking\_space  
 + room\_type\_reserved + market\_segment\_type + repeated\_guest  
 + no\_of\_previous\_cancellations + no\_of\_previous\_bookings\_not\_canceled  
 + no\_of\_special\_requests + arrival\_season + arrival\_day,  
 test\_set[, c("no\_of\_adults","no\_of\_children","no\_of\_weekend\_nights",  
 "no\_of\_week\_nights","type\_of\_meal\_plan",  
 "required\_car\_parking\_space",  
 "room\_type\_reserved","market\_segment\_type",  
 "repeated\_guest","no\_of\_previous\_cancellations",  
 "no\_of\_previous\_bookings\_not\_canceled",  
 "no\_of\_special\_requests","arrival\_season",  
 "arrival\_day")],  
 levelsOnly = F,  
 fullRank = T)  
  
onehot\_enc\_test <- predict(onehot\_encoder, test\_set[, c("no\_of\_adults","no\_of\_children","no\_of\_weekend\_nights",  
 "no\_of\_week\_nights","type\_of\_meal\_plan",  
 "required\_car\_parking\_space",  
 "room\_type\_reserved","market\_segment\_type",  
 "repeated\_guest","no\_of\_previous\_cancellations",  
 "no\_of\_previous\_bookings\_not\_canceled",  
 "no\_of\_special\_requests","arrival\_season",  
 "arrival\_day")])  
  
test\_set <- cbind(test\_set, onehot\_enc\_test)  
  
  
##### Numerical Features  
  
test\_set[, c("lead\_time", "avg\_price\_per\_room")] <- scale(test\_set[, c("lead\_time", "avg\_price\_per\_room")],  
 center = apply(training\_set[, c("lead\_time", "avg\_price\_per\_room")], 2, mean),  
 scale = apply(training\_set[, c("lead\_time", "avg\_price\_per\_room")], 2, sd))  
training\_set[, c("lead\_time", "avg\_price\_per\_room")] <- scale(training\_set[, c("lead\_time", "avg\_price\_per\_room")])  
  
  
##### Create tensors  
  
train\_col <- ncol(training\_set)  
test\_col <- ncol(test\_set)  
  
training\_features <- array(data = unlist(training\_set[, c(8,14,20:train\_col)]),  
 dim = c(nrow(training\_set), length(c(8,14,20:train\_col))))  
training\_labels <- array(data = unlist(training\_set[, "booking\_status"]),  
 dim = nrow(training\_set))  
  
test\_features <- array(data = unlist(test\_set[, c(8,14,20:test\_col)]),  
 dim = c(nrow(test\_set), length(c(8,14,20:test\_col))))  
test\_labels <- array(data = unlist(test\_set[, "booking\_status"]),  
 dim = nrow(test\_set))

## Feed-forward Dense Neural Network

### Tensorflow Workspace

library(reticulate)

## Warning: package 'reticulate' was built under R version 4.3.2

library(tensorflow)

## Warning: package 'tensorflow' was built under R version 4.3.2

##   
## Attaching package: 'tensorflow'

## The following object is masked from 'package:caret':  
##   
## train

library(keras)

## Warning: package 'keras' was built under R version 4.3.2

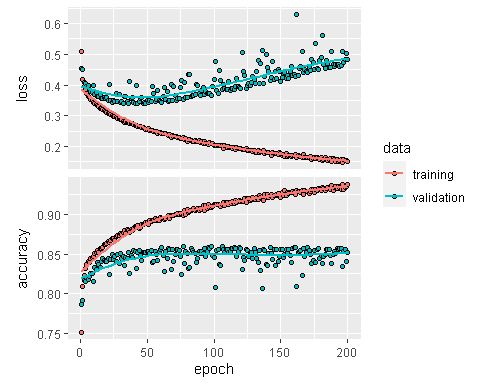
use\_virtualenv("my\_tf\_workspace")

### Overfit Model

model <- keras\_model\_sequential(list(  
 layer\_dense(units = 100, activation = "relu"),  
 layer\_dense(units = 100, activation = "relu"),  
 layer\_dense(units = 50, activation = "relu"),  
 layer\_dense(units = 1, activation = "sigmoid")  
))  
  
compile(model,  
 optimizer = "rmsprop",  
 loss = "binary\_crossentropy",  
 metrics = "accuracy")  
  
history <- fit(model, training\_features, training\_labels,  
 epochs = 200, batch\_size = 512, validation\_split = 0.33)

## Epoch 1/200  
## 36/36 - 1s - loss: 0.5093 - accuracy: 0.7514 - val\_loss: 0.4530 - val\_accuracy: 0.7857 - 825ms/epoch - 23ms/step  
## Epoch 2/200  
## 36/36 - 0s - loss: 0.4189 - accuracy: 0.8091 - val\_loss: 0.4510 - val\_accuracy: 0.7910 - 123ms/epoch - 3ms/step  
## Epoch 3/200  
## 36/36 - 0s - loss: 0.3965 - accuracy: 0.8225 - val\_loss: 0.4088 - val\_accuracy: 0.8169 - 124ms/epoch - 3ms/step  
## Epoch 4/200  
## 36/36 - 0s - loss: 0.3819 - accuracy: 0.8320 - val\_loss: 0.4048 - val\_accuracy: 0.8151 - 115ms/epoch - 3ms/step  
## Epoch 5/200  
## 36/36 - 0s - loss: 0.3740 - accuracy: 0.8363 - val\_loss: 0.3961 - val\_accuracy: 0.8198 - 118ms/epoch - 3ms/step  
## Epoch 6/200  
## 36/36 - 0s - loss: 0.3643 - accuracy: 0.8405 - val\_loss: 0.3977 - val\_accuracy: 0.8175 - 122ms/epoch - 3ms/step  
## Epoch 7/200  
## 36/36 - 0s - loss: 0.3598 - accuracy: 0.8388 - val\_loss: 0.3870 - val\_accuracy: 0.8211 - 115ms/epoch - 3ms/step  
## Epoch 8/200  
## 36/36 - 0s - loss: 0.3545 - accuracy: 0.8441 - val\_loss: 0.3787 - val\_accuracy: 0.8256 - 122ms/epoch - 3ms/step  
## Epoch 9/200  
## 36/36 - 0s - loss: 0.3498 - accuracy: 0.8452 - val\_loss: 0.3758 - val\_accuracy: 0.8258 - 120ms/epoch - 3ms/step  
## Epoch 10/200  
## 36/36 - 0s - loss: 0.3440 - accuracy: 0.8472 - val\_loss: 0.3978 - val\_accuracy: 0.8157 - 115ms/epoch - 3ms/step  
## Epoch 11/200  
## 36/36 - 0s - loss: 0.3405 - accuracy: 0.8509 - val\_loss: 0.3693 - val\_accuracy: 0.8333 - 120ms/epoch - 3ms/step  
## Epoch 12/200  
## 36/36 - 0s - loss: 0.3371 - accuracy: 0.8514 - val\_loss: 0.3753 - val\_accuracy: 0.8299 - 124ms/epoch - 3ms/step  
## Epoch 13/200  
## 36/36 - 0s - loss: 0.3325 - accuracy: 0.8546 - val\_loss: 0.3635 - val\_accuracy: 0.8390 - 121ms/epoch - 3ms/step  
## Epoch 14/200  
## 36/36 - 0s - loss: 0.3307 - accuracy: 0.8526 - val\_loss: 0.3628 - val\_accuracy: 0.8368 - 123ms/epoch - 3ms/step  
## Epoch 15/200  
## 36/36 - 0s - loss: 0.3252 - accuracy: 0.8568 - val\_loss: 0.3704 - val\_accuracy: 0.8331 - 121ms/epoch - 3ms/step  
## Epoch 16/200  
## 36/36 - 0s - loss: 0.3249 - accuracy: 0.8567 - val\_loss: 0.3572 - val\_accuracy: 0.8410 - 122ms/epoch - 3ms/step  
## Epoch 17/200  
## 36/36 - 0s - loss: 0.3216 - accuracy: 0.8577 - val\_loss: 0.3565 - val\_accuracy: 0.8444 - 136ms/epoch - 4ms/step  
## Epoch 18/200  
## 36/36 - 0s - loss: 0.3164 - accuracy: 0.8624 - val\_loss: 0.3603 - val\_accuracy: 0.8418 - 143ms/epoch - 4ms/step  
## Epoch 19/200  
## 36/36 - 0s - loss: 0.3144 - accuracy: 0.8626 - val\_loss: 0.3577 - val\_accuracy: 0.8411 - 126ms/epoch - 4ms/step  
## Epoch 20/200  
## 36/36 - 0s - loss: 0.3114 - accuracy: 0.8637 - val\_loss: 0.3603 - val\_accuracy: 0.8408 - 126ms/epoch - 4ms/step  
## Epoch 21/200  
## 36/36 - 0s - loss: 0.3083 - accuracy: 0.8656 - val\_loss: 0.3486 - val\_accuracy: 0.8447 - 120ms/epoch - 3ms/step  
## Epoch 22/200  
## 36/36 - 0s - loss: 0.3064 - accuracy: 0.8646 - val\_loss: 0.4006 - val\_accuracy: 0.8240 - 117ms/epoch - 3ms/step  
## Epoch 23/200  
## 36/36 - 0s - loss: 0.3054 - accuracy: 0.8669 - val\_loss: 0.3489 - val\_accuracy: 0.8434 - 124ms/epoch - 3ms/step  
## Epoch 24/200  
## 36/36 - 0s - loss: 0.3016 - accuracy: 0.8699 - val\_loss: 0.3442 - val\_accuracy: 0.8492 - 118ms/epoch - 3ms/step  
## Epoch 25/200  
## 36/36 - 0s - loss: 0.2976 - accuracy: 0.8693 - val\_loss: 0.3495 - val\_accuracy: 0.8434 - 121ms/epoch - 3ms/step  
## Epoch 26/200  
## 36/36 - 0s - loss: 0.2985 - accuracy: 0.8673 - val\_loss: 0.3458 - val\_accuracy: 0.8474 - 121ms/epoch - 3ms/step  
## Epoch 27/200  
## 36/36 - 0s - loss: 0.2942 - accuracy: 0.8718 - val\_loss: 0.3712 - val\_accuracy: 0.8402 - 121ms/epoch - 3ms/step  
## Epoch 28/200  
## 36/36 - 0s - loss: 0.2921 - accuracy: 0.8734 - val\_loss: 0.3472 - val\_accuracy: 0.8492 - 121ms/epoch - 3ms/step  
## Epoch 29/200  
## 36/36 - 0s - loss: 0.2908 - accuracy: 0.8729 - val\_loss: 0.3698 - val\_accuracy: 0.8378 - 119ms/epoch - 3ms/step  
## Epoch 30/200  
## 36/36 - 0s - loss: 0.2877 - accuracy: 0.8759 - val\_loss: 0.3442 - val\_accuracy: 0.8504 - 117ms/epoch - 3ms/step  
## Epoch 31/200  
## 36/36 - 0s - loss: 0.2887 - accuracy: 0.8743 - val\_loss: 0.3465 - val\_accuracy: 0.8447 - 126ms/epoch - 3ms/step  
## Epoch 32/200  
## 36/36 - 0s - loss: 0.2826 - accuracy: 0.8776 - val\_loss: 0.3413 - val\_accuracy: 0.8473 - 127ms/epoch - 4ms/step  
## Epoch 33/200  
## 36/36 - 0s - loss: 0.2793 - accuracy: 0.8795 - val\_loss: 0.3636 - val\_accuracy: 0.8361 - 129ms/epoch - 4ms/step  
## Epoch 34/200  
## 36/36 - 0s - loss: 0.2828 - accuracy: 0.8756 - val\_loss: 0.3623 - val\_accuracy: 0.8460 - 123ms/epoch - 3ms/step  
## Epoch 35/200  
## 36/36 - 0s - loss: 0.2787 - accuracy: 0.8811 - val\_loss: 0.3482 - val\_accuracy: 0.8447 - 117ms/epoch - 3ms/step  
## Epoch 36/200  
## 36/36 - 0s - loss: 0.2770 - accuracy: 0.8798 - val\_loss: 0.3397 - val\_accuracy: 0.8514 - 128ms/epoch - 4ms/step  
## Epoch 37/200  
## 36/36 - 0s - loss: 0.2762 - accuracy: 0.8773 - val\_loss: 0.4110 - val\_accuracy: 0.8256 - 125ms/epoch - 3ms/step  
## Epoch 38/200  
## 36/36 - 0s - loss: 0.2738 - accuracy: 0.8806 - val\_loss: 0.3553 - val\_accuracy: 0.8478 - 118ms/epoch - 3ms/step  
## Epoch 39/200  
## 36/36 - 0s - loss: 0.2736 - accuracy: 0.8802 - val\_loss: 0.3593 - val\_accuracy: 0.8453 - 120ms/epoch - 3ms/step  
## Epoch 40/200  
## 36/36 - 0s - loss: 0.2721 - accuracy: 0.8820 - val\_loss: 0.3436 - val\_accuracy: 0.8541 - 127ms/epoch - 4ms/step  
## Epoch 41/200  
## 36/36 - 0s - loss: 0.2685 - accuracy: 0.8843 - val\_loss: 0.3409 - val\_accuracy: 0.8535 - 118ms/epoch - 3ms/step  
## Epoch 42/200  
## 36/36 - 0s - loss: 0.2682 - accuracy: 0.8847 - val\_loss: 0.3979 - val\_accuracy: 0.8322 - 122ms/epoch - 3ms/step  
## Epoch 43/200  
## 36/36 - 0s - loss: 0.2641 - accuracy: 0.8867 - val\_loss: 0.3400 - val\_accuracy: 0.8523 - 118ms/epoch - 3ms/step  
## Epoch 44/200  
## 36/36 - 0s - loss: 0.2667 - accuracy: 0.8829 - val\_loss: 0.3461 - val\_accuracy: 0.8541 - 118ms/epoch - 3ms/step  
## Epoch 45/200  
## 36/36 - 0s - loss: 0.2667 - accuracy: 0.8838 - val\_loss: 0.3573 - val\_accuracy: 0.8480 - 127ms/epoch - 4ms/step  
## Epoch 46/200  
## 36/36 - 0s - loss: 0.2607 - accuracy: 0.8872 - val\_loss: 0.3581 - val\_accuracy: 0.8425 - 162ms/epoch - 5ms/step  
## Epoch 47/200  
## 36/36 - 0s - loss: 0.2624 - accuracy: 0.8875 - val\_loss: 0.3406 - val\_accuracy: 0.8521 - 117ms/epoch - 3ms/step  
## Epoch 48/200  
## 36/36 - 0s - loss: 0.2581 - accuracy: 0.8888 - val\_loss: 0.3427 - val\_accuracy: 0.8514 - 113ms/epoch - 3ms/step  
## Epoch 49/200  
## 36/36 - 0s - loss: 0.2571 - accuracy: 0.8896 - val\_loss: 0.3407 - val\_accuracy: 0.8524 - 121ms/epoch - 3ms/step  
## Epoch 50/200  
## 36/36 - 0s - loss: 0.2555 - accuracy: 0.8899 - val\_loss: 0.3500 - val\_accuracy: 0.8530 - 119ms/epoch - 3ms/step  
## Epoch 51/200  
## 36/36 - 0s - loss: 0.2576 - accuracy: 0.8879 - val\_loss: 0.3419 - val\_accuracy: 0.8533 - 124ms/epoch - 3ms/step  
## Epoch 52/200  
## 36/36 - 0s - loss: 0.2515 - accuracy: 0.8920 - val\_loss: 0.3510 - val\_accuracy: 0.8502 - 117ms/epoch - 3ms/step  
## Epoch 53/200  
## 36/36 - 0s - loss: 0.2519 - accuracy: 0.8909 - val\_loss: 0.3468 - val\_accuracy: 0.8541 - 128ms/epoch - 4ms/step  
## Epoch 54/200  
## 36/36 - 0s - loss: 0.2517 - accuracy: 0.8923 - val\_loss: 0.3941 - val\_accuracy: 0.8398 - 122ms/epoch - 3ms/step  
## Epoch 55/200  
## 36/36 - 0s - loss: 0.2501 - accuracy: 0.8919 - val\_loss: 0.3714 - val\_accuracy: 0.8411 - 115ms/epoch - 3ms/step  
## Epoch 56/200  
## 36/36 - 0s - loss: 0.2482 - accuracy: 0.8948 - val\_loss: 0.3452 - val\_accuracy: 0.8531 - 120ms/epoch - 3ms/step  
## Epoch 57/200  
## 36/36 - 0s - loss: 0.2495 - accuracy: 0.8921 - val\_loss: 0.3516 - val\_accuracy: 0.8495 - 121ms/epoch - 3ms/step  
## Epoch 58/200  
## 36/36 - 0s - loss: 0.2437 - accuracy: 0.8937 - val\_loss: 0.3837 - val\_accuracy: 0.8417 - 119ms/epoch - 3ms/step  
## Epoch 59/200  
## 36/36 - 0s - loss: 0.2473 - accuracy: 0.8931 - val\_loss: 0.3486 - val\_accuracy: 0.8534 - 121ms/epoch - 3ms/step  
## Epoch 60/200  
## 36/36 - 0s - loss: 0.2444 - accuracy: 0.8958 - val\_loss: 0.3475 - val\_accuracy: 0.8564 - 120ms/epoch - 3ms/step  
## Epoch 61/200  
## 36/36 - 0s - loss: 0.2417 - accuracy: 0.8949 - val\_loss: 0.3427 - val\_accuracy: 0.8583 - 132ms/epoch - 4ms/step  
## Epoch 62/200  
## 36/36 - 0s - loss: 0.2411 - accuracy: 0.8956 - val\_loss: 0.3634 - val\_accuracy: 0.8479 - 124ms/epoch - 3ms/step  
## Epoch 63/200  
## 36/36 - 0s - loss: 0.2420 - accuracy: 0.8942 - val\_loss: 0.3912 - val\_accuracy: 0.8281 - 124ms/epoch - 3ms/step  
## Epoch 64/200  
## 36/36 - 0s - loss: 0.2430 - accuracy: 0.8946 - val\_loss: 0.3516 - val\_accuracy: 0.8528 - 114ms/epoch - 3ms/step  
## Epoch 65/200  
## 36/36 - 0s - loss: 0.2383 - accuracy: 0.8971 - val\_loss: 0.3780 - val\_accuracy: 0.8439 - 118ms/epoch - 3ms/step  
## Epoch 66/200  
## 36/36 - 0s - loss: 0.2369 - accuracy: 0.8983 - val\_loss: 0.3458 - val\_accuracy: 0.8574 - 117ms/epoch - 3ms/step  
## Epoch 67/200  
## 36/36 - 0s - loss: 0.2406 - accuracy: 0.8954 - val\_loss: 0.3507 - val\_accuracy: 0.8557 - 149ms/epoch - 4ms/step  
## Epoch 68/200  
## 36/36 - 0s - loss: 0.2306 - accuracy: 0.8987 - val\_loss: 0.4013 - val\_accuracy: 0.8311 - 149ms/epoch - 4ms/step  
## Epoch 69/200  
## 36/36 - 0s - loss: 0.2361 - accuracy: 0.8986 - val\_loss: 0.3491 - val\_accuracy: 0.8552 - 124ms/epoch - 3ms/step  
## Epoch 70/200  
## 36/36 - 0s - loss: 0.2323 - accuracy: 0.8993 - val\_loss: 0.3948 - val\_accuracy: 0.8410 - 119ms/epoch - 3ms/step  
## Epoch 71/200  
## 36/36 - 0s - loss: 0.2337 - accuracy: 0.9003 - val\_loss: 0.3675 - val\_accuracy: 0.8491 - 121ms/epoch - 3ms/step  
## Epoch 72/200  
## 36/36 - 0s - loss: 0.2319 - accuracy: 0.9003 - val\_loss: 0.3509 - val\_accuracy: 0.8562 - 131ms/epoch - 4ms/step  
## Epoch 73/200  
## 36/36 - 0s - loss: 0.2283 - accuracy: 0.9007 - val\_loss: 0.3569 - val\_accuracy: 0.8553 - 115ms/epoch - 3ms/step  
## Epoch 74/200  
## 36/36 - 0s - loss: 0.2305 - accuracy: 0.8998 - val\_loss: 0.3578 - val\_accuracy: 0.8550 - 122ms/epoch - 3ms/step  
## Epoch 75/200  
## 36/36 - 0s - loss: 0.2277 - accuracy: 0.9019 - val\_loss: 0.3933 - val\_accuracy: 0.8473 - 131ms/epoch - 4ms/step  
## Epoch 76/200  
## 36/36 - 0s - loss: 0.2248 - accuracy: 0.9016 - val\_loss: 0.4149 - val\_accuracy: 0.8245 - 122ms/epoch - 3ms/step  
## Epoch 77/200  
## 36/36 - 0s - loss: 0.2305 - accuracy: 0.8987 - val\_loss: 0.3538 - val\_accuracy: 0.8594 - 128ms/epoch - 4ms/step  
## Epoch 78/200  
## 36/36 - 0s - loss: 0.2235 - accuracy: 0.9052 - val\_loss: 0.4081 - val\_accuracy: 0.8381 - 123ms/epoch - 3ms/step  
## Epoch 79/200  
## 36/36 - 0s - loss: 0.2276 - accuracy: 0.9007 - val\_loss: 0.3585 - val\_accuracy: 0.8537 - 120ms/epoch - 3ms/step  
## Epoch 80/200  
## 36/36 - 0s - loss: 0.2178 - accuracy: 0.9064 - val\_loss: 0.3605 - val\_accuracy: 0.8507 - 131ms/epoch - 4ms/step  
## Epoch 81/200  
## 36/36 - 0s - loss: 0.2235 - accuracy: 0.9023 - val\_loss: 0.4349 - val\_accuracy: 0.8318 - 121ms/epoch - 3ms/step  
## Epoch 82/200  
## 36/36 - 0s - loss: 0.2232 - accuracy: 0.9035 - val\_loss: 0.4332 - val\_accuracy: 0.8332 - 129ms/epoch - 4ms/step  
## Epoch 83/200  
## 36/36 - 0s - loss: 0.2186 - accuracy: 0.9057 - val\_loss: 0.3617 - val\_accuracy: 0.8570 - 126ms/epoch - 3ms/step  
## Epoch 84/200  
## 36/36 - 0s - loss: 0.2173 - accuracy: 0.9059 - val\_loss: 0.3655 - val\_accuracy: 0.8540 - 127ms/epoch - 4ms/step  
## Epoch 85/200  
## 36/36 - 0s - loss: 0.2172 - accuracy: 0.9071 - val\_loss: 0.3675 - val\_accuracy: 0.8515 - 123ms/epoch - 3ms/step  
## Epoch 86/200  
## 36/36 - 0s - loss: 0.2192 - accuracy: 0.9057 - val\_loss: 0.3701 - val\_accuracy: 0.8512 - 128ms/epoch - 4ms/step  
## Epoch 87/200  
## 36/36 - 0s - loss: 0.2180 - accuracy: 0.9065 - val\_loss: 0.4021 - val\_accuracy: 0.8398 - 122ms/epoch - 3ms/step  
## Epoch 88/200  
## 36/36 - 0s - loss: 0.2153 - accuracy: 0.9076 - val\_loss: 0.3997 - val\_accuracy: 0.8444 - 132ms/epoch - 4ms/step  
## Epoch 89/200  
## 36/36 - 0s - loss: 0.2195 - accuracy: 0.9053 - val\_loss: 0.3730 - val\_accuracy: 0.8530 - 121ms/epoch - 3ms/step  
## Epoch 90/200  
## 36/36 - 0s - loss: 0.2145 - accuracy: 0.9078 - val\_loss: 0.3676 - val\_accuracy: 0.8509 - 119ms/epoch - 3ms/step  
## Epoch 91/200  
## 36/36 - 0s - loss: 0.2151 - accuracy: 0.9063 - val\_loss: 0.3839 - val\_accuracy: 0.8535 - 128ms/epoch - 4ms/step  
## Epoch 92/200  
## 36/36 - 0s - loss: 0.2089 - accuracy: 0.9108 - val\_loss: 0.3915 - val\_accuracy: 0.8505 - 120ms/epoch - 3ms/step  
## Epoch 93/200  
## 36/36 - 0s - loss: 0.2161 - accuracy: 0.9052 - val\_loss: 0.3708 - val\_accuracy: 0.8570 - 120ms/epoch - 3ms/step  
## Epoch 94/200  
## 36/36 - 0s - loss: 0.2101 - accuracy: 0.9108 - val\_loss: 0.3778 - val\_accuracy: 0.8475 - 121ms/epoch - 3ms/step  
## Epoch 95/200  
## 36/36 - 0s - loss: 0.2109 - accuracy: 0.9072 - val\_loss: 0.3727 - val\_accuracy: 0.8531 - 374ms/epoch - 10ms/step  
## Epoch 96/200  
## 36/36 - 0s - loss: 0.2071 - accuracy: 0.9123 - val\_loss: 0.3710 - val\_accuracy: 0.8585 - 116ms/epoch - 3ms/step  
## Epoch 97/200  
## 36/36 - 0s - loss: 0.2120 - accuracy: 0.9086 - val\_loss: 0.3720 - val\_accuracy: 0.8569 - 123ms/epoch - 3ms/step  
## Epoch 98/200  
## 36/36 - 0s - loss: 0.2099 - accuracy: 0.9085 - val\_loss: 0.3662 - val\_accuracy: 0.8592 - 125ms/epoch - 3ms/step  
## Epoch 99/200  
## 36/36 - 0s - loss: 0.2091 - accuracy: 0.9102 - val\_loss: 0.3955 - val\_accuracy: 0.8477 - 119ms/epoch - 3ms/step  
## Epoch 100/200  
## 36/36 - 0s - loss: 0.2040 - accuracy: 0.9130 - val\_loss: 0.4177 - val\_accuracy: 0.8453 - 124ms/epoch - 3ms/step  
## Epoch 101/200  
## 36/36 - 0s - loss: 0.2023 - accuracy: 0.9124 - val\_loss: 0.4779 - val\_accuracy: 0.8081 - 117ms/epoch - 3ms/step  
## Epoch 102/200  
## 36/36 - 0s - loss: 0.2048 - accuracy: 0.9124 - val\_loss: 0.3839 - val\_accuracy: 0.8542 - 120ms/epoch - 3ms/step  
## Epoch 103/200  
## 36/36 - 0s - loss: 0.2045 - accuracy: 0.9132 - val\_loss: 0.3732 - val\_accuracy: 0.8552 - 122ms/epoch - 3ms/step  
## Epoch 104/200  
## 36/36 - 0s - loss: 0.2042 - accuracy: 0.9111 - val\_loss: 0.4465 - val\_accuracy: 0.8370 - 117ms/epoch - 3ms/step  
## Epoch 105/200  
## 36/36 - 0s - loss: 0.2046 - accuracy: 0.9106 - val\_loss: 0.4149 - val\_accuracy: 0.8360 - 127ms/epoch - 4ms/step  
## Epoch 106/200  
## 36/36 - 0s - loss: 0.2026 - accuracy: 0.9122 - val\_loss: 0.3849 - val\_accuracy: 0.8594 - 115ms/epoch - 3ms/step  
## Epoch 107/200  
## 36/36 - 0s - loss: 0.1945 - accuracy: 0.9158 - val\_loss: 0.4023 - val\_accuracy: 0.8561 - 120ms/epoch - 3ms/step  
## Epoch 108/200  
## 36/36 - 0s - loss: 0.2036 - accuracy: 0.9109 - val\_loss: 0.3889 - val\_accuracy: 0.8518 - 121ms/epoch - 3ms/step  
## Epoch 109/200  
## 36/36 - 0s - loss: 0.2007 - accuracy: 0.9151 - val\_loss: 0.3842 - val\_accuracy: 0.8559 - 121ms/epoch - 3ms/step  
## Epoch 110/200  
## 36/36 - 0s - loss: 0.1975 - accuracy: 0.9158 - val\_loss: 0.3852 - val\_accuracy: 0.8553 - 117ms/epoch - 3ms/step  
## Epoch 111/200  
## 36/36 - 0s - loss: 0.1994 - accuracy: 0.9128 - val\_loss: 0.4111 - val\_accuracy: 0.8517 - 140ms/epoch - 4ms/step  
## Epoch 112/200  
## 36/36 - 0s - loss: 0.1970 - accuracy: 0.9154 - val\_loss: 0.3820 - val\_accuracy: 0.8536 - 122ms/epoch - 3ms/step  
## Epoch 113/200  
## 36/36 - 0s - loss: 0.1979 - accuracy: 0.9138 - val\_loss: 0.3880 - val\_accuracy: 0.8565 - 120ms/epoch - 3ms/step  
## Epoch 114/200  
## 36/36 - 0s - loss: 0.1943 - accuracy: 0.9170 - val\_loss: 0.3878 - val\_accuracy: 0.8575 - 120ms/epoch - 3ms/step  
## Epoch 115/200  
## 36/36 - 0s - loss: 0.1982 - accuracy: 0.9140 - val\_loss: 0.3855 - val\_accuracy: 0.8586 - 113ms/epoch - 3ms/step  
## Epoch 116/200  
## 36/36 - 0s - loss: 0.1917 - accuracy: 0.9182 - val\_loss: 0.3943 - val\_accuracy: 0.8566 - 125ms/epoch - 3ms/step  
## Epoch 117/200  
## 36/36 - 0s - loss: 0.1964 - accuracy: 0.9150 - val\_loss: 0.4216 - val\_accuracy: 0.8474 - 119ms/epoch - 3ms/step  
## Epoch 118/200  
## 36/36 - 0s - loss: 0.1950 - accuracy: 0.9159 - val\_loss: 0.3960 - val\_accuracy: 0.8533 - 123ms/epoch - 3ms/step  
## Epoch 119/200  
## 36/36 - 0s - loss: 0.1924 - accuracy: 0.9172 - val\_loss: 0.3915 - val\_accuracy: 0.8589 - 124ms/epoch - 3ms/step  
## Epoch 120/200  
## 36/36 - 0s - loss: 0.1945 - accuracy: 0.9170 - val\_loss: 0.3891 - val\_accuracy: 0.8573 - 117ms/epoch - 3ms/step  
## Epoch 121/200  
## 36/36 - 0s - loss: 0.1894 - accuracy: 0.9167 - val\_loss: 0.5043 - val\_accuracy: 0.8309 - 121ms/epoch - 3ms/step  
## Epoch 122/200  
## 36/36 - 0s - loss: 0.1898 - accuracy: 0.9164 - val\_loss: 0.4435 - val\_accuracy: 0.8454 - 121ms/epoch - 3ms/step  
## Epoch 123/200  
## 36/36 - 0s - loss: 0.1910 - accuracy: 0.9197 - val\_loss: 0.4319 - val\_accuracy: 0.8415 - 120ms/epoch - 3ms/step  
## Epoch 124/200  
## 36/36 - 0s - loss: 0.1944 - accuracy: 0.9154 - val\_loss: 0.3903 - val\_accuracy: 0.8553 - 119ms/epoch - 3ms/step  
## Epoch 125/200  
## 36/36 - 0s - loss: 0.1867 - accuracy: 0.9204 - val\_loss: 0.3978 - val\_accuracy: 0.8544 - 118ms/epoch - 3ms/step  
## Epoch 126/200  
## 36/36 - 0s - loss: 0.1856 - accuracy: 0.9203 - val\_loss: 0.4050 - val\_accuracy: 0.8465 - 125ms/epoch - 3ms/step  
## Epoch 127/200  
## 36/36 - 0s - loss: 0.1885 - accuracy: 0.9173 - val\_loss: 0.3988 - val\_accuracy: 0.8523 - 120ms/epoch - 3ms/step  
## Epoch 128/200  
## 36/36 - 0s - loss: 0.1878 - accuracy: 0.9189 - val\_loss: 0.4075 - val\_accuracy: 0.8517 - 123ms/epoch - 3ms/step  
## Epoch 129/200  
## 36/36 - 0s - loss: 0.1842 - accuracy: 0.9212 - val\_loss: 0.4281 - val\_accuracy: 0.8439 - 121ms/epoch - 3ms/step  
## Epoch 130/200  
## 36/36 - 0s - loss: 0.1887 - accuracy: 0.9194 - val\_loss: 0.4172 - val\_accuracy: 0.8567 - 113ms/epoch - 3ms/step  
## Epoch 131/200  
## 36/36 - 0s - loss: 0.1815 - accuracy: 0.9195 - val\_loss: 0.4045 - val\_accuracy: 0.8521 - 123ms/epoch - 3ms/step  
## Epoch 132/200  
## 36/36 - 0s - loss: 0.1822 - accuracy: 0.9212 - val\_loss: 0.4431 - val\_accuracy: 0.8447 - 125ms/epoch - 3ms/step  
## Epoch 133/200  
## 36/36 - 0s - loss: 0.1870 - accuracy: 0.9187 - val\_loss: 0.4812 - val\_accuracy: 0.8398 - 116ms/epoch - 3ms/step  
## Epoch 134/200  
## 36/36 - 0s - loss: 0.1810 - accuracy: 0.9238 - val\_loss: 0.4036 - val\_accuracy: 0.8551 - 123ms/epoch - 3ms/step  
## Epoch 135/200  
## 36/36 - 0s - loss: 0.1847 - accuracy: 0.9210 - val\_loss: 0.4692 - val\_accuracy: 0.8291 - 123ms/epoch - 3ms/step  
## Epoch 136/200  
## 36/36 - 0s - loss: 0.1828 - accuracy: 0.9224 - val\_loss: 0.5117 - val\_accuracy: 0.8067 - 112ms/epoch - 3ms/step  
## Epoch 137/200  
## 36/36 - 0s - loss: 0.1889 - accuracy: 0.9173 - val\_loss: 0.4056 - val\_accuracy: 0.8564 - 126ms/epoch - 3ms/step  
## Epoch 138/200  
## 36/36 - 0s - loss: 0.1770 - accuracy: 0.9255 - val\_loss: 0.4264 - val\_accuracy: 0.8546 - 123ms/epoch - 3ms/step  
## Epoch 139/200  
## 36/36 - 0s - loss: 0.1818 - accuracy: 0.9211 - val\_loss: 0.5006 - val\_accuracy: 0.8366 - 117ms/epoch - 3ms/step  
## Epoch 140/200  
## 36/36 - 0s - loss: 0.1792 - accuracy: 0.9255 - val\_loss: 0.4563 - val\_accuracy: 0.8352 - 125ms/epoch - 3ms/step  
## Epoch 141/200  
## 36/36 - 0s - loss: 0.1804 - accuracy: 0.9211 - val\_loss: 0.4428 - val\_accuracy: 0.8511 - 122ms/epoch - 3ms/step  
## Epoch 142/200  
## 36/36 - 0s - loss: 0.1778 - accuracy: 0.9240 - val\_loss: 0.4162 - val\_accuracy: 0.8580 - 125ms/epoch - 3ms/step  
## Epoch 143/200  
## 36/36 - 0s - loss: 0.1718 - accuracy: 0.9255 - val\_loss: 0.4255 - val\_accuracy: 0.8506 - 124ms/epoch - 3ms/step  
## Epoch 144/200  
## 36/36 - 0s - loss: 0.1803 - accuracy: 0.9221 - val\_loss: 0.4758 - val\_accuracy: 0.8431 - 137ms/epoch - 4ms/step  
## Epoch 145/200  
## 36/36 - 0s - loss: 0.1772 - accuracy: 0.9241 - val\_loss: 0.4173 - val\_accuracy: 0.8553 - 124ms/epoch - 3ms/step  
## Epoch 146/200  
## 36/36 - 0s - loss: 0.1780 - accuracy: 0.9238 - val\_loss: 0.4436 - val\_accuracy: 0.8361 - 125ms/epoch - 3ms/step  
## Epoch 147/200  
## 36/36 - 0s - loss: 0.1766 - accuracy: 0.9249 - val\_loss: 0.4200 - val\_accuracy: 0.8552 - 125ms/epoch - 3ms/step  
## Epoch 148/200  
## 36/36 - 0s - loss: 0.1730 - accuracy: 0.9262 - val\_loss: 0.4494 - val\_accuracy: 0.8410 - 149ms/epoch - 4ms/step  
## Epoch 149/200  
## 36/36 - 0s - loss: 0.1794 - accuracy: 0.9226 - val\_loss: 0.4440 - val\_accuracy: 0.8473 - 146ms/epoch - 4ms/step  
## Epoch 150/200  
## 36/36 - 0s - loss: 0.1783 - accuracy: 0.9217 - val\_loss: 0.4118 - val\_accuracy: 0.8560 - 147ms/epoch - 4ms/step  
## Epoch 151/200  
## 36/36 - 0s - loss: 0.1702 - accuracy: 0.9275 - val\_loss: 0.4248 - val\_accuracy: 0.8552 - 121ms/epoch - 3ms/step  
## Epoch 152/200  
## 36/36 - 0s - loss: 0.1750 - accuracy: 0.9244 - val\_loss: 0.4386 - val\_accuracy: 0.8459 - 159ms/epoch - 4ms/step  
## Epoch 153/200  
## 36/36 - 0s - loss: 0.1690 - accuracy: 0.9298 - val\_loss: 0.4408 - val\_accuracy: 0.8489 - 143ms/epoch - 4ms/step  
## Epoch 154/200  
## 36/36 - 0s - loss: 0.1787 - accuracy: 0.9238 - val\_loss: 0.4225 - val\_accuracy: 0.8562 - 143ms/epoch - 4ms/step  
## Epoch 155/200  
## 36/36 - 0s - loss: 0.1678 - accuracy: 0.9275 - val\_loss: 0.4309 - val\_accuracy: 0.8504 - 126ms/epoch - 3ms/step  
## Epoch 156/200  
## 36/36 - 0s - loss: 0.1715 - accuracy: 0.9268 - val\_loss: 0.4204 - val\_accuracy: 0.8556 - 122ms/epoch - 3ms/step  
## Epoch 157/200  
## 36/36 - 0s - loss: 0.1698 - accuracy: 0.9277 - val\_loss: 0.4429 - val\_accuracy: 0.8483 - 153ms/epoch - 4ms/step  
## Epoch 158/200  
## 36/36 - 0s - loss: 0.1674 - accuracy: 0.9278 - val\_loss: 0.4610 - val\_accuracy: 0.8399 - 136ms/epoch - 4ms/step  
## Epoch 159/200  
## 36/36 - 0s - loss: 0.1695 - accuracy: 0.9290 - val\_loss: 0.4319 - val\_accuracy: 0.8571 - 130ms/epoch - 4ms/step  
## Epoch 160/200  
## 36/36 - 0s - loss: 0.1705 - accuracy: 0.9282 - val\_loss: 0.4346 - val\_accuracy: 0.8584 - 132ms/epoch - 4ms/step  
## Epoch 161/200  
## 36/36 - 0s - loss: 0.1736 - accuracy: 0.9255 - val\_loss: 0.4292 - val\_accuracy: 0.8577 - 122ms/epoch - 3ms/step  
## Epoch 162/200  
## 36/36 - 0s - loss: 0.1650 - accuracy: 0.9300 - val\_loss: 0.6289 - val\_accuracy: 0.8090 - 119ms/epoch - 3ms/step  
## Epoch 163/200  
## 36/36 - 0s - loss: 0.1695 - accuracy: 0.9287 - val\_loss: 0.4311 - val\_accuracy: 0.8514 - 121ms/epoch - 3ms/step  
## Epoch 164/200  
## 36/36 - 0s - loss: 0.1680 - accuracy: 0.9290 - val\_loss: 0.4554 - val\_accuracy: 0.8533 - 118ms/epoch - 3ms/step  
## Epoch 165/200  
## 36/36 - 0s - loss: 0.1668 - accuracy: 0.9277 - val\_loss: 0.4612 - val\_accuracy: 0.8551 - 124ms/epoch - 3ms/step  
## Epoch 166/200  
## 36/36 - 0s - loss: 0.1654 - accuracy: 0.9304 - val\_loss: 0.5097 - val\_accuracy: 0.8226 - 123ms/epoch - 3ms/step  
## Epoch 167/200  
## 36/36 - 0s - loss: 0.1739 - accuracy: 0.9251 - val\_loss: 0.4414 - val\_accuracy: 0.8574 - 127ms/epoch - 4ms/step  
## Epoch 168/200  
## 36/36 - 0s - loss: 0.1638 - accuracy: 0.9307 - val\_loss: 0.4498 - val\_accuracy: 0.8472 - 127ms/epoch - 4ms/step  
## Epoch 169/200  
## 36/36 - 0s - loss: 0.1660 - accuracy: 0.9297 - val\_loss: 0.4444 - val\_accuracy: 0.8523 - 120ms/epoch - 3ms/step  
## Epoch 170/200  
## 36/36 - 0s - loss: 0.1629 - accuracy: 0.9291 - val\_loss: 0.4414 - val\_accuracy: 0.8589 - 123ms/epoch - 3ms/step  
## Epoch 171/200  
## 36/36 - 0s - loss: 0.1659 - accuracy: 0.9270 - val\_loss: 0.4586 - val\_accuracy: 0.8551 - 124ms/epoch - 3ms/step  
## Epoch 172/200  
## 36/36 - 0s - loss: 0.1662 - accuracy: 0.9292 - val\_loss: 0.4488 - val\_accuracy: 0.8491 - 126ms/epoch - 4ms/step  
## Epoch 173/200  
## 36/36 - 0s - loss: 0.1587 - accuracy: 0.9321 - val\_loss: 0.4830 - val\_accuracy: 0.8376 - 121ms/epoch - 3ms/step  
## Epoch 174/200  
## 36/36 - 0s - loss: 0.1615 - accuracy: 0.9312 - val\_loss: 0.4739 - val\_accuracy: 0.8419 - 125ms/epoch - 3ms/step  
## Epoch 175/200  
## 36/36 - 0s - loss: 0.1655 - accuracy: 0.9282 - val\_loss: 0.4493 - val\_accuracy: 0.8548 - 128ms/epoch - 4ms/step  
## Epoch 176/200  
## 36/36 - 0s - loss: 0.1609 - accuracy: 0.9312 - val\_loss: 0.5351 - val\_accuracy: 0.8394 - 127ms/epoch - 4ms/step  
## Epoch 177/200  
## 36/36 - 0s - loss: 0.1648 - accuracy: 0.9303 - val\_loss: 0.4550 - val\_accuracy: 0.8557 - 124ms/epoch - 3ms/step  
## Epoch 178/200  
## 36/36 - 0s - loss: 0.1614 - accuracy: 0.9329 - val\_loss: 0.4500 - val\_accuracy: 0.8571 - 137ms/epoch - 4ms/step  
## Epoch 179/200  
## 36/36 - 0s - loss: 0.1596 - accuracy: 0.9312 - val\_loss: 0.4790 - val\_accuracy: 0.8439 - 122ms/epoch - 3ms/step  
## Epoch 180/200  
## 36/36 - 0s - loss: 0.1629 - accuracy: 0.9309 - val\_loss: 0.4894 - val\_accuracy: 0.8482 - 124ms/epoch - 3ms/step  
## Epoch 181/200  
## 36/36 - 0s - loss: 0.1561 - accuracy: 0.9343 - val\_loss: 0.5617 - val\_accuracy: 0.8386 - 118ms/epoch - 3ms/step  
## Epoch 182/200  
## 36/36 - 0s - loss: 0.1585 - accuracy: 0.9326 - val\_loss: 0.4670 - val\_accuracy: 0.8470 - 131ms/epoch - 4ms/step  
## Epoch 183/200  
## 36/36 - 0s - loss: 0.1629 - accuracy: 0.9309 - val\_loss: 0.4536 - val\_accuracy: 0.8493 - 123ms/epoch - 3ms/step  
## Epoch 184/200  
## 36/36 - 0s - loss: 0.1570 - accuracy: 0.9332 - val\_loss: 0.4794 - val\_accuracy: 0.8562 - 129ms/epoch - 4ms/step  
## Epoch 185/200  
## 36/36 - 0s - loss: 0.1562 - accuracy: 0.9339 - val\_loss: 0.4711 - val\_accuracy: 0.8562 - 116ms/epoch - 3ms/step  
## Epoch 186/200  
## 36/36 - 0s - loss: 0.1566 - accuracy: 0.9331 - val\_loss: 0.4575 - val\_accuracy: 0.8553 - 125ms/epoch - 3ms/step  
## Epoch 187/200  
## 36/36 - 0s - loss: 0.1597 - accuracy: 0.9314 - val\_loss: 0.4570 - val\_accuracy: 0.8485 - 120ms/epoch - 3ms/step  
## Epoch 188/200  
## 36/36 - 0s - loss: 0.1517 - accuracy: 0.9371 - val\_loss: 0.5102 - val\_accuracy: 0.8385 - 118ms/epoch - 3ms/step  
## Epoch 189/200  
## 36/36 - 0s - loss: 0.1615 - accuracy: 0.9302 - val\_loss: 0.4672 - val\_accuracy: 0.8491 - 143ms/epoch - 4ms/step  
## Epoch 190/200  
## 36/36 - 0s - loss: 0.1568 - accuracy: 0.9339 - val\_loss: 0.4606 - val\_accuracy: 0.8537 - 151ms/epoch - 4ms/step  
## Epoch 191/200  
## 36/36 - 0s - loss: 0.1529 - accuracy: 0.9353 - val\_loss: 0.4682 - val\_accuracy: 0.8507 - 133ms/epoch - 4ms/step  
## Epoch 192/200  
## 36/36 - 0s - loss: 0.1565 - accuracy: 0.9325 - val\_loss: 0.4668 - val\_accuracy: 0.8571 - 128ms/epoch - 4ms/step  
## Epoch 193/200  
## 36/36 - 0s - loss: 0.1524 - accuracy: 0.9358 - val\_loss: 0.4660 - val\_accuracy: 0.8560 - 117ms/epoch - 3ms/step  
## Epoch 194/200  
## 36/36 - 0s - loss: 0.1536 - accuracy: 0.9342 - val\_loss: 0.4620 - val\_accuracy: 0.8574 - 127ms/epoch - 4ms/step  
## Epoch 195/200  
## 36/36 - 0s - loss: 0.1573 - accuracy: 0.9320 - val\_loss: 0.4638 - val\_accuracy: 0.8571 - 124ms/epoch - 3ms/step  
## Epoch 196/200  
## 36/36 - 0s - loss: 0.1487 - accuracy: 0.9381 - val\_loss: 0.5101 - val\_accuracy: 0.8346 - 118ms/epoch - 3ms/step  
## Epoch 197/200  
## 36/36 - 0s - loss: 0.1564 - accuracy: 0.9319 - val\_loss: 0.4725 - val\_accuracy: 0.8534 - 123ms/epoch - 3ms/step  
## Epoch 198/200  
## 36/36 - 0s - loss: 0.1531 - accuracy: 0.9344 - val\_loss: 0.4848 - val\_accuracy: 0.8564 - 127ms/epoch - 4ms/step  
## Epoch 199/200  
## 36/36 - 0s - loss: 0.1534 - accuracy: 0.9351 - val\_loss: 0.5018 - val\_accuracy: 0.8408 - 121ms/epoch - 3ms/step  
## Epoch 200/200  
## 36/36 - 0s - loss: 0.1510 - accuracy: 0.9376 - val\_loss: 0.4819 - val\_accuracy: 0.8522 - 129ms/epoch - 4ms/step

plot(history)

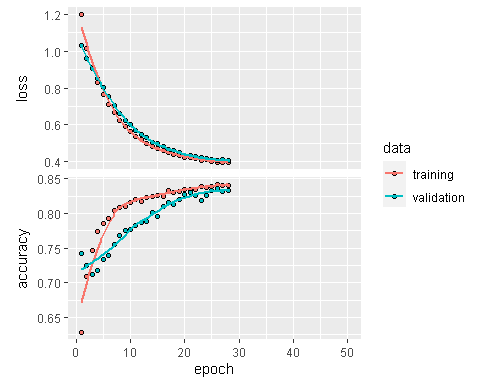


### First generalized model

model <- keras\_model\_sequential(list(  
 layer\_dense(units = 100, activation = "relu",  
 kernel\_regularizer = regularizer\_l2(0.002)),  
 layer\_batch\_normalization(),  
 layer\_dropout(rate=0.5),  
 layer\_dense(units = 100, activation = "relu",  
 kernel\_regularizer = regularizer\_l2(0.002)),  
 layer\_batch\_normalization(),  
 layer\_dropout(rate=0.5),  
 layer\_dense(units = 50, activation = "relu",  
 kernel\_regularizer = regularizer\_l2(0.002)),  
 layer\_batch\_normalization(),  
 layer\_dropout(rate=0.5),  
 layer\_dense(units = 1, activation = "sigmoid")  
))  
  
compile(model,  
 optimizer = "rmsprop",  
 loss = "binary\_crossentropy",  
 metrics = "accuracy")  
  
history <- fit(model, training\_features, training\_labels,  
 epochs = 50, batch\_size = 512, validation\_split = 0.33,  
 callbacks = list(callback\_early\_stopping(patience = 2)))

## Epoch 1/50  
## 36/36 - 1s - loss: 1.1999 - accuracy: 0.6289 - val\_loss: 1.0305 - val\_accuracy: 0.7421 - 1s/epoch - 35ms/step  
## Epoch 2/50  
## 36/36 - 0s - loss: 1.0129 - accuracy: 0.7085 - val\_loss: 0.9614 - val\_accuracy: 0.7243 - 206ms/epoch - 6ms/step  
## Epoch 3/50  
## 36/36 - 0s - loss: 0.9136 - accuracy: 0.7472 - val\_loss: 0.9061 - val\_accuracy: 0.7122 - 191ms/epoch - 5ms/step  
## Epoch 4/50  
## 36/36 - 0s - loss: 0.8297 - accuracy: 0.7746 - val\_loss: 0.8522 - val\_accuracy: 0.7182 - 202ms/epoch - 6ms/step  
## Epoch 5/50  
## 36/36 - 0s - loss: 0.7672 - accuracy: 0.7854 - val\_loss: 0.8001 - val\_accuracy: 0.7331 - 200ms/epoch - 6ms/step  
## Epoch 6/50  
## 36/36 - 0s - loss: 0.7125 - accuracy: 0.7933 - val\_loss: 0.7536 - val\_accuracy: 0.7400 - 193ms/epoch - 5ms/step  
## Epoch 7/50  
## 36/36 - 0s - loss: 0.6642 - accuracy: 0.8041 - val\_loss: 0.7038 - val\_accuracy: 0.7548 - 199ms/epoch - 6ms/step  
## Epoch 8/50  
## 36/36 - 0s - loss: 0.6227 - accuracy: 0.8087 - val\_loss: 0.6628 - val\_accuracy: 0.7689 - 199ms/epoch - 6ms/step  
## Epoch 9/50  
## 36/36 - 0s - loss: 0.5913 - accuracy: 0.8107 - val\_loss: 0.6233 - val\_accuracy: 0.7758 - 189ms/epoch - 5ms/step  
## Epoch 10/50  
## 36/36 - 0s - loss: 0.5622 - accuracy: 0.8164 - val\_loss: 0.5993 - val\_accuracy: 0.7769 - 200ms/epoch - 6ms/step  
## Epoch 11/50  
## 36/36 - 0s - loss: 0.5343 - accuracy: 0.8207 - val\_loss: 0.5687 - val\_accuracy: 0.7829 - 194ms/epoch - 5ms/step  
## Epoch 12/50  
## 36/36 - 0s - loss: 0.5170 - accuracy: 0.8176 - val\_loss: 0.5487 - val\_accuracy: 0.7875 - 227ms/epoch - 6ms/step  
## Epoch 13/50  
## 36/36 - 0s - loss: 0.4960 - accuracy: 0.8234 - val\_loss: 0.5286 - val\_accuracy: 0.7890 - 194ms/epoch - 5ms/step  
## Epoch 14/50  
## 36/36 - 0s - loss: 0.4820 - accuracy: 0.8239 - val\_loss: 0.5054 - val\_accuracy: 0.8019 - 197ms/epoch - 5ms/step  
## Epoch 15/50  
## 36/36 - 0s - loss: 0.4691 - accuracy: 0.8263 - val\_loss: 0.4972 - val\_accuracy: 0.7960 - 208ms/epoch - 6ms/step  
## Epoch 16/50  
## 36/36 - 0s - loss: 0.4578 - accuracy: 0.8245 - val\_loss: 0.4828 - val\_accuracy: 0.8104 - 192ms/epoch - 5ms/step  
## Epoch 17/50  
## 36/36 - 0s - loss: 0.4476 - accuracy: 0.8328 - val\_loss: 0.4660 - val\_accuracy: 0.8163 - 195ms/epoch - 5ms/step  
## Epoch 18/50  
## 36/36 - 0s - loss: 0.4397 - accuracy: 0.8308 - val\_loss: 0.4580 - val\_accuracy: 0.8137 - 194ms/epoch - 5ms/step  
## Epoch 19/50  
## 36/36 - 0s - loss: 0.4324 - accuracy: 0.8318 - val\_loss: 0.4472 - val\_accuracy: 0.8208 - 203ms/epoch - 6ms/step  
## Epoch 20/50  
## 36/36 - 0s - loss: 0.4236 - accuracy: 0.8348 - val\_loss: 0.4339 - val\_accuracy: 0.8269 - 198ms/epoch - 6ms/step  
## Epoch 21/50  
## 36/36 - 0s - loss: 0.4197 - accuracy: 0.8349 - val\_loss: 0.4304 - val\_accuracy: 0.8298 - 195ms/epoch - 5ms/step  
## Epoch 22/50  
## 36/36 - 0s - loss: 0.4162 - accuracy: 0.8349 - val\_loss: 0.4258 - val\_accuracy: 0.8264 - 195ms/epoch - 5ms/step  
## Epoch 23/50  
## 36/36 - 0s - loss: 0.4059 - accuracy: 0.8393 - val\_loss: 0.4215 - val\_accuracy: 0.8195 - 191ms/epoch - 5ms/step  
## Epoch 24/50  
## 36/36 - 0s - loss: 0.4075 - accuracy: 0.8371 - val\_loss: 0.4153 - val\_accuracy: 0.8262 - 194ms/epoch - 5ms/step  
## Epoch 25/50  
## 36/36 - 0s - loss: 0.4003 - accuracy: 0.8394 - val\_loss: 0.4091 - val\_accuracy: 0.8329 - 211ms/epoch - 6ms/step  
## Epoch 26/50  
## 36/36 - 0s - loss: 0.3954 - accuracy: 0.8420 - val\_loss: 0.4027 - val\_accuracy: 0.8359 - 193ms/epoch - 5ms/step  
## Epoch 27/50  
## 36/36 - 0s - loss: 0.3945 - accuracy: 0.8408 - val\_loss: 0.4081 - val\_accuracy: 0.8323 - 199ms/epoch - 6ms/step  
## Epoch 28/50  
## 36/36 - 0s - loss: 0.3952 - accuracy: 0.8408 - val\_loss: 0.4042 - val\_accuracy: 0.8338 - 207ms/epoch - 6ms/step

plot(history)



### Evaluate generalized model

predictions <- predict(model, test\_features)

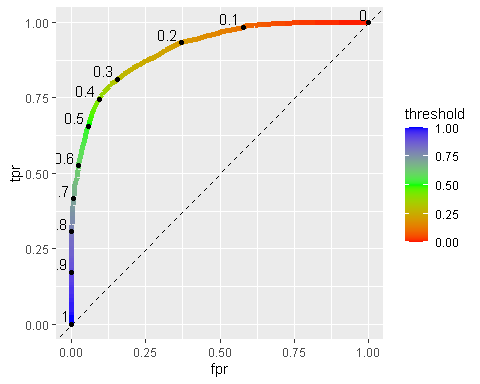
## 284/284 - 0s - 295ms/epoch - 1ms/step

test\_set$p\_prob <- predictions[, 1]

#### ROC curve

roc\_data <- data.frame(threshold=seq(1,0,-0.01), fpr=0, tpr=0)  
for (i in roc\_data$threshold) {  
   
 over\_threshold <- test\_set[test\_set$p\_prob >= i, ]  
   
 fpr <- sum(over\_threshold$booking\_status==0)/sum(test\_set$booking\_status==0)  
 roc\_data[roc\_data$threshold==i, "fpr"] <- fpr  
   
 tpr <- sum(over\_threshold$booking\_status==1)/sum(test\_set$booking\_status==1)  
 roc\_data[roc\_data$threshold==i, "tpr"] <- tpr  
   
}  
  
ggplot() +  
 geom\_line(data = roc\_data, aes(x=fpr, y=tpr, color = threshold), size = 2) +  
 scale\_color\_gradientn(colors = rainbow(3)) +  
 geom\_abline(intercept = 0, slope = 1, lty = 2) +  
 geom\_point(data = roc\_data[seq(1, 101, 10), ], aes(x = fpr, y =tpr)) +  
 geom\_text(data = roc\_data[seq(1, 101, 10), ],  
 aes(x = fpr, y = tpr, label = threshold, hjust = 1.2, vjust = -0.2))

## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.  
## ℹ Please use `linewidth` instead.  
## This warning is displayed once every 8 hours.  
## Call `lifecycle::last\_lifecycle\_warnings()` to see where this warning was  
## generated.



#### AUC

auc <- auc(x = roc\_data$fpr, y = roc\_data$tpr, type = "spline")

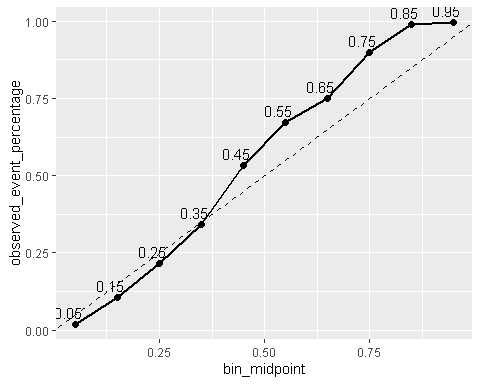
## Warning in regularize.values(x, y, ties, missing(ties)): collapsing to unique  
## 'x' values

auc

## [1] 0.9125296

#### Calibration Curve

calibration\_data <- data.frame(bin\_midpoint=seq(0.05, 0.95, 0.1),  
 observed\_event\_percentage=0)  
for (i in seq(0.05,0.95,0.1)) {  
   
 in\_interval <- test\_set[test\_set$p\_prob >= (i-0.05) & test\_set$p\_prob <= (i+0.05), ]  
 oep <- nrow(in\_interval[in\_interval$booking\_status==1, ])/nrow(in\_interval)  
 calibration\_data[calibration\_data$bin\_midpoint==i, "observed\_event\_percentage"] <- oep  
   
}  
  
ggplot(data = calibration\_data, aes(x = bin\_midpoint, y = observed\_event\_percentage)) +  
 geom\_line(size = 1) +  
 geom\_abline(intercept = 0, slope = 1, lty = 2) +  
 geom\_point(size = 2) +  
 geom\_text(aes(label = bin\_midpoint), hjust = 0.75, vjust = -0.5)

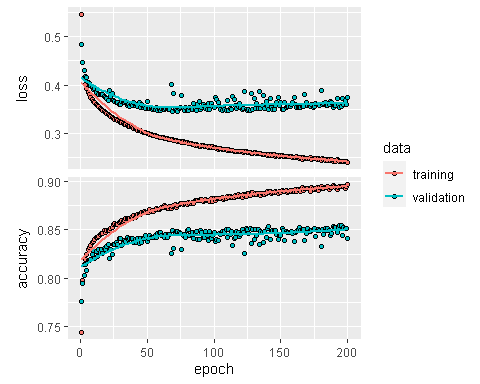


### Model that Fails to Overfit

model\_small <- keras\_model\_sequential(list(  
 layer\_dense(units = 75, activation = "relu"),  
 layer\_dense(units = 37, activation = "relu"),  
 layer\_dense(units = 1, activation = "sigmoid")  
))  
  
compile(model\_small,  
 optimizer = "rmsprop",  
 loss = "binary\_crossentropy",  
 metrics = "accuracy")  
  
history\_small <- fit(model\_small, training\_features, training\_labels,  
 epochs = 200, batch\_size = 512, validation\_split = 0.33)

## Epoch 1/200  
## 36/36 - 1s - loss: 0.5458 - accuracy: 0.7441 - val\_loss: 0.4845 - val\_accuracy: 0.7763 - 592ms/epoch - 16ms/step  
## Epoch 2/200  
## 36/36 - 0s - loss: 0.4476 - accuracy: 0.7971 - val\_loss: 0.4471 - val\_accuracy: 0.7949 - 104ms/epoch - 3ms/step  
## Epoch 3/200  
## 36/36 - 0s - loss: 0.4151 - accuracy: 0.8145 - val\_loss: 0.4295 - val\_accuracy: 0.8031 - 99ms/epoch - 3ms/step  
## Epoch 4/200  
## 36/36 - 0s - loss: 0.4018 - accuracy: 0.8196 - val\_loss: 0.4183 - val\_accuracy: 0.8142 - 103ms/epoch - 3ms/step  
## Epoch 5/200  
## 36/36 - 0s - loss: 0.3922 - accuracy: 0.8245 - val\_loss: 0.4156 - val\_accuracy: 0.8079 - 100ms/epoch - 3ms/step  
## Epoch 6/200  
## 36/36 - 0s - loss: 0.3855 - accuracy: 0.8298 - val\_loss: 0.4060 - val\_accuracy: 0.8184 - 96ms/epoch - 3ms/step  
## Epoch 7/200  
## 36/36 - 0s - loss: 0.3800 - accuracy: 0.8320 - val\_loss: 0.4032 - val\_accuracy: 0.8188 - 101ms/epoch - 3ms/step  
## Epoch 8/200  
## 36/36 - 0s - loss: 0.3747 - accuracy: 0.8337 - val\_loss: 0.3999 - val\_accuracy: 0.8193 - 98ms/epoch - 3ms/step  
## Epoch 9/200  
## 36/36 - 0s - loss: 0.3705 - accuracy: 0.8375 - val\_loss: 0.3945 - val\_accuracy: 0.8243 - 96ms/epoch - 3ms/step  
## Epoch 10/200  
## 36/36 - 0s - loss: 0.3667 - accuracy: 0.8383 - val\_loss: 0.3910 - val\_accuracy: 0.8287 - 95ms/epoch - 3ms/step  
## Epoch 11/200  
## 36/36 - 0s - loss: 0.3639 - accuracy: 0.8405 - val\_loss: 0.3936 - val\_accuracy: 0.8201 - 99ms/epoch - 3ms/step  
## Epoch 12/200  
## 36/36 - 0s - loss: 0.3602 - accuracy: 0.8417 - val\_loss: 0.3856 - val\_accuracy: 0.8281 - 97ms/epoch - 3ms/step  
## Epoch 13/200  
## 36/36 - 0s - loss: 0.3578 - accuracy: 0.8420 - val\_loss: 0.3901 - val\_accuracy: 0.8227 - 108ms/epoch - 3ms/step  
## Epoch 14/200  
## 36/36 - 0s - loss: 0.3552 - accuracy: 0.8428 - val\_loss: 0.3811 - val\_accuracy: 0.8301 - 100ms/epoch - 3ms/step  
## Epoch 15/200  
## 36/36 - 0s - loss: 0.3522 - accuracy: 0.8448 - val\_loss: 0.3812 - val\_accuracy: 0.8265 - 105ms/epoch - 3ms/step  
## Epoch 16/200  
## 36/36 - 0s - loss: 0.3494 - accuracy: 0.8463 - val\_loss: 0.3802 - val\_accuracy: 0.8295 - 101ms/epoch - 3ms/step  
## Epoch 17/200  
## 36/36 - 0s - loss: 0.3470 - accuracy: 0.8467 - val\_loss: 0.3809 - val\_accuracy: 0.8259 - 94ms/epoch - 3ms/step  
## Epoch 18/200  
## 36/36 - 0s - loss: 0.3448 - accuracy: 0.8499 - val\_loss: 0.3765 - val\_accuracy: 0.8309 - 101ms/epoch - 3ms/step  
## Epoch 19/200  
## 36/36 - 0s - loss: 0.3433 - accuracy: 0.8477 - val\_loss: 0.3746 - val\_accuracy: 0.8299 - 97ms/epoch - 3ms/step  
## Epoch 20/200  
## 36/36 - 0s - loss: 0.3410 - accuracy: 0.8490 - val\_loss: 0.3712 - val\_accuracy: 0.8330 - 98ms/epoch - 3ms/step  
## Epoch 21/200  
## 36/36 - 0s - loss: 0.3386 - accuracy: 0.8496 - val\_loss: 0.3690 - val\_accuracy: 0.8352 - 102ms/epoch - 3ms/step  
## Epoch 22/200  
## 36/36 - 0s - loss: 0.3374 - accuracy: 0.8495 - val\_loss: 0.3881 - val\_accuracy: 0.8206 - 101ms/epoch - 3ms/step  
## Epoch 23/200  
## 36/36 - 0s - loss: 0.3352 - accuracy: 0.8519 - val\_loss: 0.3767 - val\_accuracy: 0.8295 - 101ms/epoch - 3ms/step  
## Epoch 24/200  
## 36/36 - 0s - loss: 0.3327 - accuracy: 0.8524 - val\_loss: 0.3818 - val\_accuracy: 0.8249 - 96ms/epoch - 3ms/step  
## Epoch 25/200  
## 36/36 - 0s - loss: 0.3308 - accuracy: 0.8552 - val\_loss: 0.3678 - val\_accuracy: 0.8348 - 111ms/epoch - 3ms/step  
## Epoch 26/200  
## 36/36 - 0s - loss: 0.3294 - accuracy: 0.8564 - val\_loss: 0.3652 - val\_accuracy: 0.8352 - 100ms/epoch - 3ms/step  
## Epoch 27/200  
## 36/36 - 0s - loss: 0.3283 - accuracy: 0.8554 - val\_loss: 0.3635 - val\_accuracy: 0.8368 - 98ms/epoch - 3ms/step  
## Epoch 28/200  
## 36/36 - 0s - loss: 0.3265 - accuracy: 0.8557 - val\_loss: 0.3643 - val\_accuracy: 0.8347 - 102ms/epoch - 3ms/step  
## Epoch 29/200  
## 36/36 - 0s - loss: 0.3244 - accuracy: 0.8574 - val\_loss: 0.3603 - val\_accuracy: 0.8409 - 103ms/epoch - 3ms/step  
## Epoch 30/200  
## 36/36 - 0s - loss: 0.3233 - accuracy: 0.8585 - val\_loss: 0.3604 - val\_accuracy: 0.8408 - 95ms/epoch - 3ms/step  
## Epoch 31/200  
## 36/36 - 0s - loss: 0.3218 - accuracy: 0.8598 - val\_loss: 0.3655 - val\_accuracy: 0.8342 - 101ms/epoch - 3ms/step  
## Epoch 32/200  
## 36/36 - 0s - loss: 0.3204 - accuracy: 0.8589 - val\_loss: 0.3632 - val\_accuracy: 0.8367 - 99ms/epoch - 3ms/step  
## Epoch 33/200  
## 36/36 - 0s - loss: 0.3185 - accuracy: 0.8600 - val\_loss: 0.3648 - val\_accuracy: 0.8387 - 104ms/epoch - 3ms/step  
## Epoch 34/200  
## 36/36 - 0s - loss: 0.3182 - accuracy: 0.8614 - val\_loss: 0.3649 - val\_accuracy: 0.8378 - 99ms/epoch - 3ms/step  
## Epoch 35/200  
## 36/36 - 0s - loss: 0.3159 - accuracy: 0.8619 - val\_loss: 0.3587 - val\_accuracy: 0.8423 - 103ms/epoch - 3ms/step  
## Epoch 36/200  
## 36/36 - 0s - loss: 0.3144 - accuracy: 0.8619 - val\_loss: 0.3588 - val\_accuracy: 0.8401 - 99ms/epoch - 3ms/step  
## Epoch 37/200  
## 36/36 - 0s - loss: 0.3135 - accuracy: 0.8623 - val\_loss: 0.3547 - val\_accuracy: 0.8414 - 98ms/epoch - 3ms/step  
## Epoch 38/200  
## 36/36 - 0s - loss: 0.3131 - accuracy: 0.8628 - val\_loss: 0.3588 - val\_accuracy: 0.8411 - 98ms/epoch - 3ms/step  
## Epoch 39/200  
## 36/36 - 0s - loss: 0.3122 - accuracy: 0.8624 - val\_loss: 0.3571 - val\_accuracy: 0.8416 - 101ms/epoch - 3ms/step  
## Epoch 40/200  
## 36/36 - 0s - loss: 0.3090 - accuracy: 0.8650 - val\_loss: 0.3634 - val\_accuracy: 0.8372 - 95ms/epoch - 3ms/step  
## Epoch 41/200  
## 36/36 - 0s - loss: 0.3091 - accuracy: 0.8633 - val\_loss: 0.3583 - val\_accuracy: 0.8402 - 92ms/epoch - 3ms/step  
## Epoch 42/200  
## 36/36 - 0s - loss: 0.3078 - accuracy: 0.8654 - val\_loss: 0.3592 - val\_accuracy: 0.8369 - 99ms/epoch - 3ms/step  
## Epoch 43/200  
## 36/36 - 0s - loss: 0.3066 - accuracy: 0.8661 - val\_loss: 0.3554 - val\_accuracy: 0.8412 - 96ms/epoch - 3ms/step  
## Epoch 44/200  
## 36/36 - 0s - loss: 0.3066 - accuracy: 0.8659 - val\_loss: 0.3637 - val\_accuracy: 0.8360 - 96ms/epoch - 3ms/step  
## Epoch 45/200  
## 36/36 - 0s - loss: 0.3052 - accuracy: 0.8658 - val\_loss: 0.3526 - val\_accuracy: 0.8410 - 104ms/epoch - 3ms/step  
## Epoch 46/200  
## 36/36 - 0s - loss: 0.3030 - accuracy: 0.8679 - val\_loss: 0.3596 - val\_accuracy: 0.8367 - 104ms/epoch - 3ms/step  
## Epoch 47/200  
## 36/36 - 0s - loss: 0.3026 - accuracy: 0.8680 - val\_loss: 0.3543 - val\_accuracy: 0.8410 - 97ms/epoch - 3ms/step  
## Epoch 48/200  
## 36/36 - 0s - loss: 0.3015 - accuracy: 0.8687 - val\_loss: 0.3553 - val\_accuracy: 0.8417 - 107ms/epoch - 3ms/step  
## Epoch 49/200  
## 36/36 - 0s - loss: 0.3008 - accuracy: 0.8682 - val\_loss: 0.3519 - val\_accuracy: 0.8439 - 97ms/epoch - 3ms/step  
## Epoch 50/200  
## 36/36 - 0s - loss: 0.2997 - accuracy: 0.8681 - val\_loss: 0.3649 - val\_accuracy: 0.8382 - 98ms/epoch - 3ms/step  
## Epoch 51/200  
## 36/36 - 0s - loss: 0.2997 - accuracy: 0.8681 - val\_loss: 0.3523 - val\_accuracy: 0.8415 - 112ms/epoch - 3ms/step  
## Epoch 52/200  
## 36/36 - 0s - loss: 0.2973 - accuracy: 0.8717 - val\_loss: 0.3515 - val\_accuracy: 0.8443 - 99ms/epoch - 3ms/step  
## Epoch 53/200  
## 36/36 - 0s - loss: 0.2973 - accuracy: 0.8690 - val\_loss: 0.3516 - val\_accuracy: 0.8425 - 94ms/epoch - 3ms/step  
## Epoch 54/200  
## 36/36 - 0s - loss: 0.2962 - accuracy: 0.8706 - val\_loss: 0.3495 - val\_accuracy: 0.8425 - 98ms/epoch - 3ms/step  
## Epoch 55/200  
## 36/36 - 0s - loss: 0.2950 - accuracy: 0.8694 - val\_loss: 0.3506 - val\_accuracy: 0.8419 - 102ms/epoch - 3ms/step  
## Epoch 56/200  
## 36/36 - 0s - loss: 0.2947 - accuracy: 0.8711 - val\_loss: 0.3486 - val\_accuracy: 0.8441 - 98ms/epoch - 3ms/step  
## Epoch 57/200  
## 36/36 - 0s - loss: 0.2939 - accuracy: 0.8720 - val\_loss: 0.3530 - val\_accuracy: 0.8420 - 103ms/epoch - 3ms/step  
## Epoch 58/200  
## 36/36 - 0s - loss: 0.2940 - accuracy: 0.8708 - val\_loss: 0.3481 - val\_accuracy: 0.8460 - 93ms/epoch - 3ms/step  
## Epoch 59/200  
## 36/36 - 0s - loss: 0.2919 - accuracy: 0.8715 - val\_loss: 0.3488 - val\_accuracy: 0.8477 - 96ms/epoch - 3ms/step  
## Epoch 60/200  
## 36/36 - 0s - loss: 0.2913 - accuracy: 0.8725 - val\_loss: 0.3481 - val\_accuracy: 0.8477 - 100ms/epoch - 3ms/step  
## Epoch 61/200  
## 36/36 - 0s - loss: 0.2923 - accuracy: 0.8729 - val\_loss: 0.3523 - val\_accuracy: 0.8438 - 95ms/epoch - 3ms/step  
## Epoch 62/200  
## 36/36 - 0s - loss: 0.2905 - accuracy: 0.8736 - val\_loss: 0.3495 - val\_accuracy: 0.8437 - 93ms/epoch - 3ms/step  
## Epoch 63/200  
## 36/36 - 0s - loss: 0.2886 - accuracy: 0.8751 - val\_loss: 0.3493 - val\_accuracy: 0.8425 - 100ms/epoch - 3ms/step  
## Epoch 64/200  
## 36/36 - 0s - loss: 0.2897 - accuracy: 0.8742 - val\_loss: 0.3505 - val\_accuracy: 0.8448 - 98ms/epoch - 3ms/step  
## Epoch 65/200  
## 36/36 - 0s - loss: 0.2890 - accuracy: 0.8719 - val\_loss: 0.3524 - val\_accuracy: 0.8426 - 95ms/epoch - 3ms/step  
## Epoch 66/200  
## 36/36 - 0s - loss: 0.2877 - accuracy: 0.8722 - val\_loss: 0.3491 - val\_accuracy: 0.8453 - 103ms/epoch - 3ms/step  
## Epoch 67/200  
## 36/36 - 0s - loss: 0.2877 - accuracy: 0.8740 - val\_loss: 0.3503 - val\_accuracy: 0.8450 - 97ms/epoch - 3ms/step  
## Epoch 68/200  
## 36/36 - 0s - loss: 0.2861 - accuracy: 0.8724 - val\_loss: 0.4006 - val\_accuracy: 0.8258 - 96ms/epoch - 3ms/step  
## Epoch 69/200  
## 36/36 - 0s - loss: 0.2864 - accuracy: 0.8743 - val\_loss: 0.3456 - val\_accuracy: 0.8491 - 98ms/epoch - 3ms/step  
## Epoch 70/200  
## 36/36 - 0s - loss: 0.2859 - accuracy: 0.8753 - val\_loss: 0.3833 - val\_accuracy: 0.8312 - 101ms/epoch - 3ms/step  
## Epoch 71/200  
## 36/36 - 0s - loss: 0.2841 - accuracy: 0.8763 - val\_loss: 0.3489 - val\_accuracy: 0.8475 - 97ms/epoch - 3ms/step  
## Epoch 72/200  
## 36/36 - 0s - loss: 0.2847 - accuracy: 0.8738 - val\_loss: 0.3470 - val\_accuracy: 0.8488 - 102ms/epoch - 3ms/step  
## Epoch 73/200  
## 36/36 - 0s - loss: 0.2834 - accuracy: 0.8757 - val\_loss: 0.3462 - val\_accuracy: 0.8482 - 98ms/epoch - 3ms/step  
## Epoch 74/200  
## 36/36 - 0s - loss: 0.2823 - accuracy: 0.8769 - val\_loss: 0.3511 - val\_accuracy: 0.8478 - 99ms/epoch - 3ms/step  
## Epoch 75/200  
## 36/36 - 0s - loss: 0.2823 - accuracy: 0.8776 - val\_loss: 0.3527 - val\_accuracy: 0.8459 - 102ms/epoch - 3ms/step  
## Epoch 76/200  
## 36/36 - 0s - loss: 0.2815 - accuracy: 0.8770 - val\_loss: 0.3784 - val\_accuracy: 0.8302 - 102ms/epoch - 3ms/step  
## Epoch 77/200  
## 36/36 - 0s - loss: 0.2824 - accuracy: 0.8760 - val\_loss: 0.3506 - val\_accuracy: 0.8446 - 97ms/epoch - 3ms/step  
## Epoch 78/200  
## 36/36 - 0s - loss: 0.2809 - accuracy: 0.8773 - val\_loss: 0.3475 - val\_accuracy: 0.8450 - 111ms/epoch - 3ms/step  
## Epoch 79/200  
## 36/36 - 0s - loss: 0.2808 - accuracy: 0.8778 - val\_loss: 0.3532 - val\_accuracy: 0.8426 - 106ms/epoch - 3ms/step  
## Epoch 80/200  
## 36/36 - 0s - loss: 0.2804 - accuracy: 0.8767 - val\_loss: 0.3465 - val\_accuracy: 0.8491 - 97ms/epoch - 3ms/step  
## Epoch 81/200  
## 36/36 - 0s - loss: 0.2784 - accuracy: 0.8787 - val\_loss: 0.3463 - val\_accuracy: 0.8505 - 94ms/epoch - 3ms/step  
## Epoch 82/200  
## 36/36 - 0s - loss: 0.2782 - accuracy: 0.8775 - val\_loss: 0.3545 - val\_accuracy: 0.8440 - 96ms/epoch - 3ms/step  
## Epoch 83/200  
## 36/36 - 0s - loss: 0.2782 - accuracy: 0.8772 - val\_loss: 0.3512 - val\_accuracy: 0.8457 - 96ms/epoch - 3ms/step  
## Epoch 84/200  
## 36/36 - 0s - loss: 0.2770 - accuracy: 0.8785 - val\_loss: 0.3489 - val\_accuracy: 0.8475 - 99ms/epoch - 3ms/step  
## Epoch 85/200  
## 36/36 - 0s - loss: 0.2770 - accuracy: 0.8776 - val\_loss: 0.3609 - val\_accuracy: 0.8411 - 96ms/epoch - 3ms/step  
## Epoch 86/200  
## 36/36 - 0s - loss: 0.2751 - accuracy: 0.8804 - val\_loss: 0.3486 - val\_accuracy: 0.8478 - 104ms/epoch - 3ms/step  
## Epoch 87/200  
## 36/36 - 0s - loss: 0.2763 - accuracy: 0.8804 - val\_loss: 0.3470 - val\_accuracy: 0.8485 - 98ms/epoch - 3ms/step  
## Epoch 88/200  
## 36/36 - 0s - loss: 0.2748 - accuracy: 0.8802 - val\_loss: 0.3603 - val\_accuracy: 0.8406 - 99ms/epoch - 3ms/step  
## Epoch 89/200  
## 36/36 - 0s - loss: 0.2735 - accuracy: 0.8802 - val\_loss: 0.3662 - val\_accuracy: 0.8401 - 99ms/epoch - 3ms/step  
## Epoch 90/200  
## 36/36 - 0s - loss: 0.2735 - accuracy: 0.8818 - val\_loss: 0.3496 - val\_accuracy: 0.8498 - 101ms/epoch - 3ms/step  
## Epoch 91/200  
## 36/36 - 0s - loss: 0.2743 - accuracy: 0.8814 - val\_loss: 0.3543 - val\_accuracy: 0.8473 - 92ms/epoch - 3ms/step  
## Epoch 92/200  
## 36/36 - 0s - loss: 0.2742 - accuracy: 0.8799 - val\_loss: 0.3590 - val\_accuracy: 0.8423 - 95ms/epoch - 3ms/step  
## Epoch 93/200  
## 36/36 - 0s - loss: 0.2728 - accuracy: 0.8804 - val\_loss: 0.3559 - val\_accuracy: 0.8427 - 99ms/epoch - 3ms/step  
## Epoch 94/200  
## 36/36 - 0s - loss: 0.2724 - accuracy: 0.8812 - val\_loss: 0.3518 - val\_accuracy: 0.8488 - 96ms/epoch - 3ms/step  
## Epoch 95/200  
## 36/36 - 0s - loss: 0.2727 - accuracy: 0.8804 - val\_loss: 0.3503 - val\_accuracy: 0.8441 - 99ms/epoch - 3ms/step  
## Epoch 96/200  
## 36/36 - 0s - loss: 0.2700 - accuracy: 0.8809 - val\_loss: 0.3626 - val\_accuracy: 0.8387 - 110ms/epoch - 3ms/step  
## Epoch 97/200  
## 36/36 - 0s - loss: 0.2724 - accuracy: 0.8797 - val\_loss: 0.3496 - val\_accuracy: 0.8486 - 98ms/epoch - 3ms/step  
## Epoch 98/200  
## 36/36 - 0s - loss: 0.2700 - accuracy: 0.8824 - val\_loss: 0.3625 - val\_accuracy: 0.8410 - 99ms/epoch - 3ms/step  
## Epoch 99/200  
## 36/36 - 0s - loss: 0.2735 - accuracy: 0.8797 - val\_loss: 0.3555 - val\_accuracy: 0.8447 - 102ms/epoch - 3ms/step  
## Epoch 100/200  
## 36/36 - 0s - loss: 0.2696 - accuracy: 0.8824 - val\_loss: 0.3535 - val\_accuracy: 0.8473 - 98ms/epoch - 3ms/step  
## Epoch 101/200  
## 36/36 - 0s - loss: 0.2702 - accuracy: 0.8807 - val\_loss: 0.3735 - val\_accuracy: 0.8380 - 95ms/epoch - 3ms/step  
## Epoch 102/200  
## 36/36 - 0s - loss: 0.2690 - accuracy: 0.8825 - val\_loss: 0.3705 - val\_accuracy: 0.8400 - 105ms/epoch - 3ms/step  
## Epoch 103/200  
## 36/36 - 0s - loss: 0.2673 - accuracy: 0.8824 - val\_loss: 0.3640 - val\_accuracy: 0.8395 - 96ms/epoch - 3ms/step  
## Epoch 104/200  
## 36/36 - 0s - loss: 0.2684 - accuracy: 0.8825 - val\_loss: 0.3551 - val\_accuracy: 0.8466 - 97ms/epoch - 3ms/step  
## Epoch 105/200  
## 36/36 - 0s - loss: 0.2694 - accuracy: 0.8825 - val\_loss: 0.3495 - val\_accuracy: 0.8458 - 103ms/epoch - 3ms/step  
## Epoch 106/200  
## 36/36 - 0s - loss: 0.2662 - accuracy: 0.8841 - val\_loss: 0.3467 - val\_accuracy: 0.8488 - 99ms/epoch - 3ms/step  
## Epoch 107/200  
## 36/36 - 0s - loss: 0.2674 - accuracy: 0.8820 - val\_loss: 0.3535 - val\_accuracy: 0.8468 - 101ms/epoch - 3ms/step  
## Epoch 108/200  
## 36/36 - 0s - loss: 0.2666 - accuracy: 0.8830 - val\_loss: 0.3497 - val\_accuracy: 0.8484 - 100ms/epoch - 3ms/step  
## Epoch 109/200  
## 36/36 - 0s - loss: 0.2670 - accuracy: 0.8827 - val\_loss: 0.3475 - val\_accuracy: 0.8507 - 95ms/epoch - 3ms/step  
## Epoch 110/200  
## 36/36 - 0s - loss: 0.2649 - accuracy: 0.8847 - val\_loss: 0.3684 - val\_accuracy: 0.8423 - 97ms/epoch - 3ms/step  
## Epoch 111/200  
## 36/36 - 0s - loss: 0.2669 - accuracy: 0.8837 - val\_loss: 0.3482 - val\_accuracy: 0.8503 - 102ms/epoch - 3ms/step  
## Epoch 112/200  
## 36/36 - 0s - loss: 0.2647 - accuracy: 0.8846 - val\_loss: 0.3504 - val\_accuracy: 0.8450 - 97ms/epoch - 3ms/step  
## Epoch 113/200  
## 36/36 - 0s - loss: 0.2651 - accuracy: 0.8837 - val\_loss: 0.3486 - val\_accuracy: 0.8485 - 97ms/epoch - 3ms/step  
## Epoch 114/200  
## 36/36 - 0s - loss: 0.2647 - accuracy: 0.8839 - val\_loss: 0.3766 - val\_accuracy: 0.8343 - 98ms/epoch - 3ms/step  
## Epoch 115/200  
## 36/36 - 0s - loss: 0.2632 - accuracy: 0.8841 - val\_loss: 0.3703 - val\_accuracy: 0.8411 - 102ms/epoch - 3ms/step  
## Epoch 116/200  
## 36/36 - 0s - loss: 0.2623 - accuracy: 0.8862 - val\_loss: 0.3568 - val\_accuracy: 0.8424 - 97ms/epoch - 3ms/step  
## Epoch 117/200  
## 36/36 - 0s - loss: 0.2632 - accuracy: 0.8852 - val\_loss: 0.3506 - val\_accuracy: 0.8482 - 135ms/epoch - 4ms/step  
## Epoch 118/200  
## 36/36 - 0s - loss: 0.2622 - accuracy: 0.8865 - val\_loss: 0.3549 - val\_accuracy: 0.8475 - 104ms/epoch - 3ms/step  
## Epoch 119/200  
## 36/36 - 0s - loss: 0.2627 - accuracy: 0.8852 - val\_loss: 0.3608 - val\_accuracy: 0.8429 - 97ms/epoch - 3ms/step  
## Epoch 120/200  
## 36/36 - 0s - loss: 0.2626 - accuracy: 0.8862 - val\_loss: 0.3528 - val\_accuracy: 0.8458 - 99ms/epoch - 3ms/step  
## Epoch 121/200  
## 36/36 - 0s - loss: 0.2613 - accuracy: 0.8837 - val\_loss: 0.3492 - val\_accuracy: 0.8505 - 98ms/epoch - 3ms/step  
## Epoch 122/200  
## 36/36 - 0s - loss: 0.2601 - accuracy: 0.8864 - val\_loss: 0.3518 - val\_accuracy: 0.8475 - 97ms/epoch - 3ms/step  
## Epoch 123/200  
## 36/36 - 0s - loss: 0.2613 - accuracy: 0.8854 - val\_loss: 0.3876 - val\_accuracy: 0.8258 - 97ms/epoch - 3ms/step  
## Epoch 124/200  
## 36/36 - 0s - loss: 0.2619 - accuracy: 0.8849 - val\_loss: 0.3503 - val\_accuracy: 0.8511 - 93ms/epoch - 3ms/step  
## Epoch 125/200  
## 36/36 - 0s - loss: 0.2597 - accuracy: 0.8874 - val\_loss: 0.3481 - val\_accuracy: 0.8509 - 99ms/epoch - 3ms/step  
## Epoch 126/200  
## 36/36 - 0s - loss: 0.2609 - accuracy: 0.8879 - val\_loss: 0.3498 - val\_accuracy: 0.8460 - 97ms/epoch - 3ms/step  
## Epoch 127/200  
## 36/36 - 0s - loss: 0.2606 - accuracy: 0.8863 - val\_loss: 0.3469 - val\_accuracy: 0.8520 - 109ms/epoch - 3ms/step  
## Epoch 128/200  
## 36/36 - 0s - loss: 0.2580 - accuracy: 0.8884 - val\_loss: 0.3824 - val\_accuracy: 0.8359 - 103ms/epoch - 3ms/step  
## Epoch 129/200  
## 36/36 - 0s - loss: 0.2588 - accuracy: 0.8889 - val\_loss: 0.3504 - val\_accuracy: 0.8503 - 100ms/epoch - 3ms/step  
## Epoch 130/200  
## 36/36 - 0s - loss: 0.2595 - accuracy: 0.8869 - val\_loss: 0.3731 - val\_accuracy: 0.8397 - 99ms/epoch - 3ms/step  
## Epoch 131/200  
## 36/36 - 0s - loss: 0.2595 - accuracy: 0.8874 - val\_loss: 0.3505 - val\_accuracy: 0.8504 - 109ms/epoch - 3ms/step  
## Epoch 132/200  
## 36/36 - 0s - loss: 0.2576 - accuracy: 0.8864 - val\_loss: 0.3623 - val\_accuracy: 0.8446 - 98ms/epoch - 3ms/step  
## Epoch 133/200  
## 36/36 - 0s - loss: 0.2582 - accuracy: 0.8877 - val\_loss: 0.3506 - val\_accuracy: 0.8509 - 99ms/epoch - 3ms/step  
## Epoch 134/200  
## 36/36 - 0s - loss: 0.2580 - accuracy: 0.8880 - val\_loss: 0.3863 - val\_accuracy: 0.8352 - 96ms/epoch - 3ms/step  
## Epoch 135/200  
## 36/36 - 0s - loss: 0.2572 - accuracy: 0.8860 - val\_loss: 0.3515 - val\_accuracy: 0.8486 - 96ms/epoch - 3ms/step  
## Epoch 136/200  
## 36/36 - 0s - loss: 0.2562 - accuracy: 0.8879 - val\_loss: 0.3574 - val\_accuracy: 0.8472 - 103ms/epoch - 3ms/step  
## Epoch 137/200  
## 36/36 - 0s - loss: 0.2554 - accuracy: 0.8877 - val\_loss: 0.3780 - val\_accuracy: 0.8373 - 100ms/epoch - 3ms/step  
## Epoch 138/200  
## 36/36 - 0s - loss: 0.2570 - accuracy: 0.8873 - val\_loss: 0.3507 - val\_accuracy: 0.8483 - 98ms/epoch - 3ms/step  
## Epoch 139/200  
## 36/36 - 0s - loss: 0.2554 - accuracy: 0.8880 - val\_loss: 0.3665 - val\_accuracy: 0.8423 - 98ms/epoch - 3ms/step  
## Epoch 140/200  
## 36/36 - 0s - loss: 0.2549 - accuracy: 0.8896 - val\_loss: 0.3551 - val\_accuracy: 0.8514 - 98ms/epoch - 3ms/step  
## Epoch 141/200  
## 36/36 - 0s - loss: 0.2553 - accuracy: 0.8890 - val\_loss: 0.3496 - val\_accuracy: 0.8533 - 94ms/epoch - 3ms/step  
## Epoch 142/200  
## 36/36 - 0s - loss: 0.2545 - accuracy: 0.8895 - val\_loss: 0.3717 - val\_accuracy: 0.8406 - 103ms/epoch - 3ms/step  
## Epoch 143/200  
## 36/36 - 0s - loss: 0.2553 - accuracy: 0.8881 - val\_loss: 0.3530 - val\_accuracy: 0.8480 - 98ms/epoch - 3ms/step  
## Epoch 144/200  
## 36/36 - 0s - loss: 0.2546 - accuracy: 0.8890 - val\_loss: 0.3502 - val\_accuracy: 0.8496 - 99ms/epoch - 3ms/step  
## Epoch 145/200  
## 36/36 - 0s - loss: 0.2551 - accuracy: 0.8890 - val\_loss: 0.3491 - val\_accuracy: 0.8499 - 111ms/epoch - 3ms/step  
## Epoch 146/200  
## 36/36 - 0s - loss: 0.2527 - accuracy: 0.8898 - val\_loss: 0.3572 - val\_accuracy: 0.8492 - 101ms/epoch - 3ms/step  
## Epoch 147/200  
## 36/36 - 0s - loss: 0.2550 - accuracy: 0.8885 - val\_loss: 0.3640 - val\_accuracy: 0.8452 - 109ms/epoch - 3ms/step  
## Epoch 148/200  
## 36/36 - 0s - loss: 0.2540 - accuracy: 0.8876 - val\_loss: 0.3743 - val\_accuracy: 0.8397 - 104ms/epoch - 3ms/step  
## Epoch 149/200  
## 36/36 - 0s - loss: 0.2528 - accuracy: 0.8900 - val\_loss: 0.3586 - val\_accuracy: 0.8443 - 94ms/epoch - 3ms/step  
## Epoch 150/200  
## 36/36 - 0s - loss: 0.2534 - accuracy: 0.8886 - val\_loss: 0.3640 - val\_accuracy: 0.8445 - 111ms/epoch - 3ms/step  
## Epoch 151/200  
## 36/36 - 0s - loss: 0.2531 - accuracy: 0.8913 - val\_loss: 0.3498 - val\_accuracy: 0.8516 - 96ms/epoch - 3ms/step  
## Epoch 152/200  
## 36/36 - 0s - loss: 0.2528 - accuracy: 0.8897 - val\_loss: 0.3500 - val\_accuracy: 0.8497 - 98ms/epoch - 3ms/step  
## Epoch 153/200  
## 36/36 - 0s - loss: 0.2513 - accuracy: 0.8896 - val\_loss: 0.3648 - val\_accuracy: 0.8394 - 101ms/epoch - 3ms/step  
## Epoch 154/200  
## 36/36 - 0s - loss: 0.2510 - accuracy: 0.8914 - val\_loss: 0.3540 - val\_accuracy: 0.8486 - 90ms/epoch - 2ms/step  
## Epoch 155/200  
## 36/36 - 0s - loss: 0.2521 - accuracy: 0.8893 - val\_loss: 0.3548 - val\_accuracy: 0.8472 - 102ms/epoch - 3ms/step  
## Epoch 156/200  
## 36/36 - 0s - loss: 0.2514 - accuracy: 0.8899 - val\_loss: 0.3560 - val\_accuracy: 0.8469 - 120ms/epoch - 3ms/step  
## Epoch 157/200  
## 36/36 - 0s - loss: 0.2519 - accuracy: 0.8907 - val\_loss: 0.3557 - val\_accuracy: 0.8473 - 103ms/epoch - 3ms/step  
## Epoch 158/200  
## 36/36 - 0s - loss: 0.2513 - accuracy: 0.8896 - val\_loss: 0.3594 - val\_accuracy: 0.8483 - 97ms/epoch - 3ms/step  
## Epoch 159/200  
## 36/36 - 0s - loss: 0.2505 - accuracy: 0.8911 - val\_loss: 0.3641 - val\_accuracy: 0.8467 - 99ms/epoch - 3ms/step  
## Epoch 160/200  
## 36/36 - 0s - loss: 0.2504 - accuracy: 0.8902 - val\_loss: 0.3747 - val\_accuracy: 0.8381 - 96ms/epoch - 3ms/step  
## Epoch 161/200  
## 36/36 - 0s - loss: 0.2493 - accuracy: 0.8918 - val\_loss: 0.3593 - val\_accuracy: 0.8478 - 103ms/epoch - 3ms/step  
## Epoch 162/200  
## 36/36 - 0s - loss: 0.2509 - accuracy: 0.8901 - val\_loss: 0.3545 - val\_accuracy: 0.8486 - 100ms/epoch - 3ms/step  
## Epoch 163/200  
## 36/36 - 0s - loss: 0.2502 - accuracy: 0.8919 - val\_loss: 0.3529 - val\_accuracy: 0.8508 - 94ms/epoch - 3ms/step  
## Epoch 164/200  
## 36/36 - 0s - loss: 0.2479 - accuracy: 0.8911 - val\_loss: 0.3538 - val\_accuracy: 0.8522 - 103ms/epoch - 3ms/step  
## Epoch 165/200  
## 36/36 - 0s - loss: 0.2493 - accuracy: 0.8895 - val\_loss: 0.3543 - val\_accuracy: 0.8499 - 100ms/epoch - 3ms/step  
## Epoch 166/200  
## 36/36 - 0s - loss: 0.2462 - accuracy: 0.8937 - val\_loss: 0.3562 - val\_accuracy: 0.8501 - 98ms/epoch - 3ms/step  
## Epoch 167/200  
## 36/36 - 0s - loss: 0.2493 - accuracy: 0.8911 - val\_loss: 0.3552 - val\_accuracy: 0.8503 - 106ms/epoch - 3ms/step  
## Epoch 168/200  
## 36/36 - 0s - loss: 0.2480 - accuracy: 0.8932 - val\_loss: 0.3807 - val\_accuracy: 0.8396 - 97ms/epoch - 3ms/step  
## Epoch 169/200  
## 36/36 - 0s - loss: 0.2488 - accuracy: 0.8932 - val\_loss: 0.3628 - val\_accuracy: 0.8459 - 98ms/epoch - 3ms/step  
## Epoch 170/200  
## 36/36 - 0s - loss: 0.2473 - accuracy: 0.8929 - val\_loss: 0.3586 - val\_accuracy: 0.8511 - 106ms/epoch - 3ms/step  
## Epoch 171/200  
## 36/36 - 0s - loss: 0.2473 - accuracy: 0.8921 - val\_loss: 0.3581 - val\_accuracy: 0.8479 - 96ms/epoch - 3ms/step  
## Epoch 172/200  
## 36/36 - 0s - loss: 0.2497 - accuracy: 0.8902 - val\_loss: 0.3546 - val\_accuracy: 0.8486 - 96ms/epoch - 3ms/step  
## Epoch 173/200  
## 36/36 - 0s - loss: 0.2455 - accuracy: 0.8938 - val\_loss: 0.3602 - val\_accuracy: 0.8498 - 103ms/epoch - 3ms/step  
## Epoch 174/200  
## 36/36 - 0s - loss: 0.2484 - accuracy: 0.8933 - val\_loss: 0.3529 - val\_accuracy: 0.8503 - 96ms/epoch - 3ms/step  
## Epoch 175/200  
## 36/36 - 0s - loss: 0.2470 - accuracy: 0.8923 - val\_loss: 0.3546 - val\_accuracy: 0.8514 - 96ms/epoch - 3ms/step  
## Epoch 176/200  
## 36/36 - 0s - loss: 0.2472 - accuracy: 0.8932 - val\_loss: 0.3609 - val\_accuracy: 0.8433 - 107ms/epoch - 3ms/step  
## Epoch 177/200  
## 36/36 - 0s - loss: 0.2457 - accuracy: 0.8938 - val\_loss: 0.3600 - val\_accuracy: 0.8495 - 99ms/epoch - 3ms/step  
## Epoch 178/200  
## 36/36 - 0s - loss: 0.2480 - accuracy: 0.8905 - val\_loss: 0.3590 - val\_accuracy: 0.8503 - 103ms/epoch - 3ms/step  
## Epoch 179/200  
## 36/36 - 0s - loss: 0.2455 - accuracy: 0.8937 - val\_loss: 0.3568 - val\_accuracy: 0.8493 - 102ms/epoch - 3ms/step  
## Epoch 180/200  
## 36/36 - 0s - loss: 0.2442 - accuracy: 0.8947 - val\_loss: 0.3884 - val\_accuracy: 0.8327 - 97ms/epoch - 3ms/step  
## Epoch 181/200  
## 36/36 - 0s - loss: 0.2472 - accuracy: 0.8929 - val\_loss: 0.3604 - val\_accuracy: 0.8504 - 100ms/epoch - 3ms/step  
## Epoch 182/200  
## 36/36 - 0s - loss: 0.2451 - accuracy: 0.8929 - val\_loss: 0.3608 - val\_accuracy: 0.8453 - 95ms/epoch - 3ms/step  
## Epoch 183/200  
## 36/36 - 0s - loss: 0.2443 - accuracy: 0.8936 - val\_loss: 0.3641 - val\_accuracy: 0.8512 - 97ms/epoch - 3ms/step  
## Epoch 184/200  
## 36/36 - 0s - loss: 0.2447 - accuracy: 0.8941 - val\_loss: 0.3568 - val\_accuracy: 0.8497 - 94ms/epoch - 3ms/step  
## Epoch 185/200  
## 36/36 - 0s - loss: 0.2433 - accuracy: 0.8951 - val\_loss: 0.3570 - val\_accuracy: 0.8512 - 100ms/epoch - 3ms/step  
## Epoch 186/200  
## 36/36 - 0s - loss: 0.2447 - accuracy: 0.8918 - val\_loss: 0.3589 - val\_accuracy: 0.8497 - 97ms/epoch - 3ms/step  
## Epoch 187/200  
## 36/36 - 0s - loss: 0.2430 - accuracy: 0.8938 - val\_loss: 0.3661 - val\_accuracy: 0.8445 - 93ms/epoch - 3ms/step  
## Epoch 188/200  
## 36/36 - 0s - loss: 0.2456 - accuracy: 0.8939 - val\_loss: 0.3633 - val\_accuracy: 0.8464 - 104ms/epoch - 3ms/step  
## Epoch 189/200  
## 36/36 - 0s - loss: 0.2417 - accuracy: 0.8962 - val\_loss: 0.3540 - val\_accuracy: 0.8541 - 97ms/epoch - 3ms/step  
## Epoch 190/200  
## 36/36 - 0s - loss: 0.2439 - accuracy: 0.8950 - val\_loss: 0.3583 - val\_accuracy: 0.8521 - 95ms/epoch - 3ms/step  
## Epoch 191/200  
## 36/36 - 0s - loss: 0.2414 - accuracy: 0.8945 - val\_loss: 0.3574 - val\_accuracy: 0.8523 - 96ms/epoch - 3ms/step  
## Epoch 192/200  
## 36/36 - 0s - loss: 0.2449 - accuracy: 0.8922 - val\_loss: 0.3716 - val\_accuracy: 0.8472 - 103ms/epoch - 3ms/step  
## Epoch 193/200  
## 36/36 - 0s - loss: 0.2434 - accuracy: 0.8944 - val\_loss: 0.3596 - val\_accuracy: 0.8521 - 93ms/epoch - 3ms/step  
## Epoch 194/200  
## 36/36 - 0s - loss: 0.2409 - accuracy: 0.8948 - val\_loss: 0.3531 - val\_accuracy: 0.8545 - 100ms/epoch - 3ms/step  
## Epoch 195/200  
## 36/36 - 0s - loss: 0.2424 - accuracy: 0.8952 - val\_loss: 0.3559 - val\_accuracy: 0.8542 - 97ms/epoch - 3ms/step  
## Epoch 196/200  
## 36/36 - 0s - loss: 0.2413 - accuracy: 0.8952 - val\_loss: 0.3712 - val\_accuracy: 0.8441 - 99ms/epoch - 3ms/step  
## Epoch 197/200  
## 36/36 - 0s - loss: 0.2409 - accuracy: 0.8946 - val\_loss: 0.3591 - val\_accuracy: 0.8521 - 114ms/epoch - 3ms/step  
## Epoch 198/200  
## 36/36 - 0s - loss: 0.2422 - accuracy: 0.8933 - val\_loss: 0.3671 - val\_accuracy: 0.8516 - 100ms/epoch - 3ms/step  
## Epoch 199/200  
## 36/36 - 0s - loss: 0.2415 - accuracy: 0.8955 - val\_loss: 0.3601 - val\_accuracy: 0.8520 - 92ms/epoch - 3ms/step  
## Epoch 200/200  
## 36/36 - 0s - loss: 0.2398 - accuracy: 0.8972 - val\_loss: 0.3746 - val\_accuracy: 0.8410 - 98ms/epoch - 3ms/step

plot(history\_small)

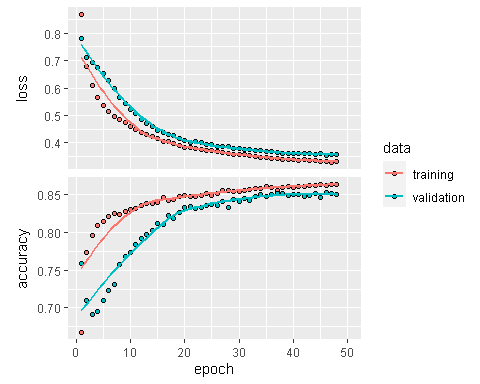


### Generalize smaller model

model\_small <- keras\_model\_sequential(list(  
 layer\_dense(units = 75, activation = "relu",  
 kernel\_regularizer = regularizer\_l2(0.002)),  
 layer\_batch\_normalization(),  
 layer\_dropout(rate=0.2),  
 layer\_dense(units = 37, activation = "relu",  
 kernel\_regularizer = regularizer\_l2(0.002)),  
 layer\_batch\_normalization(),  
 layer\_dropout(rate=0.2),  
 layer\_dense(units = 1, activation = "sigmoid")  
))  
  
compile(model\_small,  
 optimizer = "rmsprop",  
 loss = "binary\_crossentropy",  
 metrics = "accuracy")  
  
history <- fit(model\_small, training\_features, training\_labels,  
 epochs = 50, batch\_size = 512, validation\_split = 0.33,  
 callbacks = list(callback\_early\_stopping(patience = 2)))

## Epoch 1/50  
## 36/36 - 1s - loss: 0.8677 - accuracy: 0.6678 - val\_loss: 0.7795 - val\_accuracy: 0.7584 - 988ms/epoch - 27ms/step  
## Epoch 2/50  
## 36/36 - 0s - loss: 0.6787 - accuracy: 0.7738 - val\_loss: 0.7125 - val\_accuracy: 0.7101 - 145ms/epoch - 4ms/step  
## Epoch 3/50  
## 36/36 - 0s - loss: 0.6107 - accuracy: 0.7964 - val\_loss: 0.6928 - val\_accuracy: 0.6916 - 159ms/epoch - 4ms/step  
## Epoch 4/50  
## 36/36 - 0s - loss: 0.5674 - accuracy: 0.8100 - val\_loss: 0.6769 - val\_accuracy: 0.6950 - 152ms/epoch - 4ms/step  
## Epoch 5/50  
## 36/36 - 0s - loss: 0.5379 - accuracy: 0.8148 - val\_loss: 0.6536 - val\_accuracy: 0.7093 - 147ms/epoch - 4ms/step  
## Epoch 6/50  
## 36/36 - 0s - loss: 0.5132 - accuracy: 0.8214 - val\_loss: 0.6286 - val\_accuracy: 0.7232 - 145ms/epoch - 4ms/step  
## Epoch 7/50  
## 36/36 - 0s - loss: 0.4976 - accuracy: 0.8253 - val\_loss: 0.5986 - val\_accuracy: 0.7312 - 146ms/epoch - 4ms/step  
## Epoch 8/50  
## 36/36 - 0s - loss: 0.4863 - accuracy: 0.8242 - val\_loss: 0.5641 - val\_accuracy: 0.7571 - 148ms/epoch - 4ms/step  
## Epoch 9/50  
## 36/36 - 0s - loss: 0.4728 - accuracy: 0.8273 - val\_loss: 0.5439 - val\_accuracy: 0.7681 - 146ms/epoch - 4ms/step  
## Epoch 10/50  
## 36/36 - 0s - loss: 0.4609 - accuracy: 0.8310 - val\_loss: 0.5225 - val\_accuracy: 0.7730 - 161ms/epoch - 4ms/step  
## Epoch 11/50  
## 36/36 - 0s - loss: 0.4487 - accuracy: 0.8317 - val\_loss: 0.5056 - val\_accuracy: 0.7838 - 161ms/epoch - 4ms/step  
## Epoch 12/50  
## 36/36 - 0s - loss: 0.4389 - accuracy: 0.8359 - val\_loss: 0.4847 - val\_accuracy: 0.7924 - 149ms/epoch - 4ms/step  
## Epoch 13/50  
## 36/36 - 0s - loss: 0.4322 - accuracy: 0.8385 - val\_loss: 0.4694 - val\_accuracy: 0.7974 - 147ms/epoch - 4ms/step  
## Epoch 14/50  
## 36/36 - 0s - loss: 0.4243 - accuracy: 0.8380 - val\_loss: 0.4594 - val\_accuracy: 0.8026 - 143ms/epoch - 4ms/step  
## Epoch 15/50  
## 36/36 - 0s - loss: 0.4163 - accuracy: 0.8399 - val\_loss: 0.4452 - val\_accuracy: 0.8124 - 151ms/epoch - 4ms/step  
## Epoch 16/50  
## 36/36 - 0s - loss: 0.4068 - accuracy: 0.8467 - val\_loss: 0.4388 - val\_accuracy: 0.8108 - 149ms/epoch - 4ms/step  
## Epoch 17/50  
## 36/36 - 0s - loss: 0.4052 - accuracy: 0.8427 - val\_loss: 0.4295 - val\_accuracy: 0.8221 - 153ms/epoch - 4ms/step  
## Epoch 18/50  
## 36/36 - 0s - loss: 0.3994 - accuracy: 0.8443 - val\_loss: 0.4287 - val\_accuracy: 0.8190 - 158ms/epoch - 4ms/step  
## Epoch 19/50  
## 36/36 - 0s - loss: 0.3923 - accuracy: 0.8477 - val\_loss: 0.4177 - val\_accuracy: 0.8266 - 144ms/epoch - 4ms/step  
## Epoch 20/50  
## 36/36 - 0s - loss: 0.3847 - accuracy: 0.8494 - val\_loss: 0.4104 - val\_accuracy: 0.8337 - 164ms/epoch - 5ms/step  
## Epoch 21/50  
## 36/36 - 0s - loss: 0.3835 - accuracy: 0.8475 - val\_loss: 0.4028 - val\_accuracy: 0.8352 - 142ms/epoch - 4ms/step  
## Epoch 22/50  
## 36/36 - 0s - loss: 0.3787 - accuracy: 0.8479 - val\_loss: 0.4044 - val\_accuracy: 0.8319 - 146ms/epoch - 4ms/step  
## Epoch 23/50  
## 36/36 - 0s - loss: 0.3771 - accuracy: 0.8489 - val\_loss: 0.4014 - val\_accuracy: 0.8336 - 158ms/epoch - 4ms/step  
## Epoch 24/50  
## 36/36 - 0s - loss: 0.3723 - accuracy: 0.8521 - val\_loss: 0.3931 - val\_accuracy: 0.8365 - 153ms/epoch - 4ms/step  
## Epoch 25/50  
## 36/36 - 0s - loss: 0.3723 - accuracy: 0.8491 - val\_loss: 0.3928 - val\_accuracy: 0.8375 - 157ms/epoch - 4ms/step  
## Epoch 26/50  
## 36/36 - 0s - loss: 0.3698 - accuracy: 0.8519 - val\_loss: 0.3877 - val\_accuracy: 0.8356 - 145ms/epoch - 4ms/step  
## Epoch 27/50  
## 36/36 - 0s - loss: 0.3637 - accuracy: 0.8553 - val\_loss: 0.3855 - val\_accuracy: 0.8409 - 146ms/epoch - 4ms/step  
## Epoch 28/50  
## 36/36 - 0s - loss: 0.3632 - accuracy: 0.8555 - val\_loss: 0.3856 - val\_accuracy: 0.8338 - 146ms/epoch - 4ms/step  
## Epoch 29/50  
## 36/36 - 0s - loss: 0.3588 - accuracy: 0.8540 - val\_loss: 0.3791 - val\_accuracy: 0.8443 - 146ms/epoch - 4ms/step  
## Epoch 30/50  
## 36/36 - 0s - loss: 0.3579 - accuracy: 0.8541 - val\_loss: 0.3788 - val\_accuracy: 0.8416 - 144ms/epoch - 4ms/step  
## Epoch 31/50  
## 36/36 - 0s - loss: 0.3588 - accuracy: 0.8561 - val\_loss: 0.3759 - val\_accuracy: 0.8455 - 149ms/epoch - 4ms/step  
## Epoch 32/50  
## 36/36 - 0s - loss: 0.3536 - accuracy: 0.8568 - val\_loss: 0.3749 - val\_accuracy: 0.8433 - 142ms/epoch - 4ms/step  
## Epoch 33/50  
## 36/36 - 0s - loss: 0.3498 - accuracy: 0.8591 - val\_loss: 0.3709 - val\_accuracy: 0.8479 - 142ms/epoch - 4ms/step  
## Epoch 34/50  
## 36/36 - 0s - loss: 0.3474 - accuracy: 0.8586 - val\_loss: 0.3710 - val\_accuracy: 0.8512 - 145ms/epoch - 4ms/step  
## Epoch 35/50  
## 36/36 - 0s - loss: 0.3451 - accuracy: 0.8613 - val\_loss: 0.3679 - val\_accuracy: 0.8480 - 148ms/epoch - 4ms/step  
## Epoch 36/50  
## 36/36 - 0s - loss: 0.3426 - accuracy: 0.8601 - val\_loss: 0.3673 - val\_accuracy: 0.8509 - 147ms/epoch - 4ms/step  
## Epoch 37/50  
## 36/36 - 0s - loss: 0.3440 - accuracy: 0.8565 - val\_loss: 0.3644 - val\_accuracy: 0.8525 - 153ms/epoch - 4ms/step  
## Epoch 38/50  
## 36/36 - 0s - loss: 0.3398 - accuracy: 0.8604 - val\_loss: 0.3617 - val\_accuracy: 0.8520 - 157ms/epoch - 4ms/step  
## Epoch 39/50  
## 36/36 - 0s - loss: 0.3409 - accuracy: 0.8606 - val\_loss: 0.3632 - val\_accuracy: 0.8488 - 151ms/epoch - 4ms/step  
## Epoch 40/50  
## 36/36 - 0s - loss: 0.3396 - accuracy: 0.8604 - val\_loss: 0.3604 - val\_accuracy: 0.8505 - 143ms/epoch - 4ms/step  
## Epoch 41/50  
## 36/36 - 0s - loss: 0.3369 - accuracy: 0.8609 - val\_loss: 0.3603 - val\_accuracy: 0.8504 - 149ms/epoch - 4ms/step  
## Epoch 42/50  
## 36/36 - 0s - loss: 0.3381 - accuracy: 0.8612 - val\_loss: 0.3623 - val\_accuracy: 0.8473 - 145ms/epoch - 4ms/step  
## Epoch 43/50  
## 36/36 - 0s - loss: 0.3376 - accuracy: 0.8623 - val\_loss: 0.3603 - val\_accuracy: 0.8492 - 152ms/epoch - 4ms/step  
## Epoch 44/50  
## 36/36 - 0s - loss: 0.3344 - accuracy: 0.8630 - val\_loss: 0.3566 - val\_accuracy: 0.8514 - 154ms/epoch - 4ms/step  
## Epoch 45/50  
## 36/36 - 0s - loss: 0.3313 - accuracy: 0.8640 - val\_loss: 0.3628 - val\_accuracy: 0.8463 - 154ms/epoch - 4ms/step  
## Epoch 46/50  
## 36/36 - 0s - loss: 0.3306 - accuracy: 0.8628 - val\_loss: 0.3541 - val\_accuracy: 0.8538 - 143ms/epoch - 4ms/step  
## Epoch 47/50  
## 36/36 - 0s - loss: 0.3292 - accuracy: 0.8637 - val\_loss: 0.3586 - val\_accuracy: 0.8517 - 145ms/epoch - 4ms/step  
## Epoch 48/50  
## 36/36 - 0s - loss: 0.3339 - accuracy: 0.8636 - val\_loss: 0.3566 - val\_accuracy: 0.8509 - 150ms/epoch - 4ms/step

plot(history)

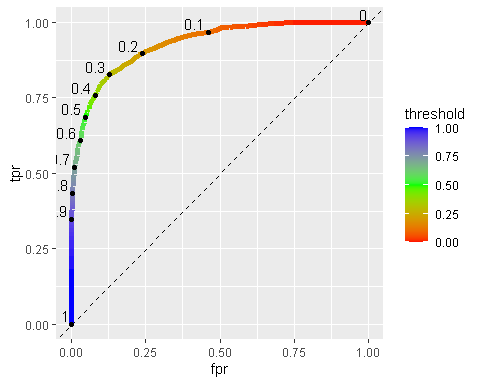


### Evaluate smaller model

# Evaluate the smaller capacity model  
  
predictions <- predict(model\_small, test\_features)

## 284/284 - 0s - 257ms/epoch - 905us/step

test\_set$p\_prob2 <- predictions[, 1]  
  
# ROC curve  
  
roc\_data <- data.frame(threshold=seq(1,0,-0.01), fpr=0, tpr=0)  
for (i in roc\_data$threshold) {  
   
 over\_threshold <- test\_set[test\_set$p\_prob2 >= i, ]  
   
 fpr <- sum(over\_threshold$booking\_status==0)/sum(test\_set$booking\_status==0)  
 roc\_data[roc\_data$threshold==i, "fpr"] <- fpr  
   
 tpr <- sum(over\_threshold$booking\_status==1)/sum(test\_set$booking\_status==1)  
 roc\_data[roc\_data$threshold==i, "tpr"] <- tpr  
   
}  
  
ggplot() +  
 geom\_line(data = roc\_data, aes(x=fpr, y=tpr, color = threshold), size = 2) +  
 scale\_color\_gradientn(colors = rainbow(3)) +  
 geom\_abline(intercept = 0, slope = 1, lty = 2) +  
 geom\_point(data = roc\_data[seq(1, 101, 10), ], aes(x = fpr, y =tpr)) +  
 geom\_text(data = roc\_data[seq(1, 101, 10), ],  
 aes(x = fpr, y = tpr, label = threshold, hjust = 1.2, vjust = -0.2))



# AUC  
  
auc <- auc(x = roc\_data$fpr, y = roc\_data$tpr, type = "spline")

## Warning in regularize.values(x, y, ties, missing(ties)): collapsing to unique  
## 'x' values

auc

## [1] 0.9284658

# Calibration curve  
  
calibration\_data <- data.frame(bin\_midpoint=seq(0.05, 0.95, 0.1),  
 observed\_event\_percentage=0)  
for (i in seq(0.05,0.95,0.1)) {  
   
 in\_interval <- test\_set[test\_set$p\_prob2 >= (i-0.05) & test\_set$p\_prob2 <= (i+0.05), ]  
 oep <- nrow(in\_interval[in\_interval$booking\_status==1, ])/nrow(in\_interval)  
 calibration\_data[calibration\_data$bin\_midpoint==i, "observed\_event\_percentage"] <- oep  
   
}  
  
ggplot(data = calibration\_data, aes(x = bin\_midpoint, y = observed\_event\_percentage)) +  
 geom\_line(size = 1) +  
 geom\_abline(intercept = 0, slope = 1, lty = 2) +  
 geom\_point(size = 2) +  
 geom\_text(aes(label = bin\_midpoint), hjust = 0.75, vjust = -0.5)

