> eval <- evaluate(model, test\_data, test\_targets, verbose = 1)

423/423 [==============================] - 1s 2ms/step - loss: 2.8729 - categorical\_accuracy: 0.6751

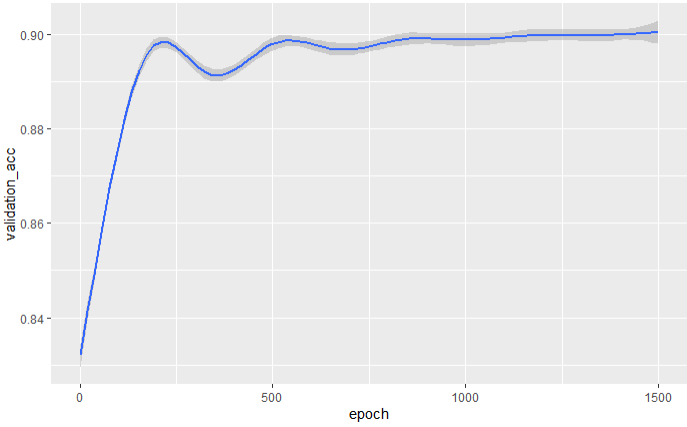
> eval

loss categorical\_accuracy

2.872923 0.675146

> max(average\_acc\_history$validation\_acc)

[1] 0.9025788



Model:

build\_model <- function() {

model <- keras\_model\_sequential() %>%

#layer\_batch\_normalization(axis = -1L, input\_shape = dim(train\_data)[[2]]) %>%

layer\_dense(units = 64, activation = "relu", input\_shape = dim(train\_data)[[2]]) %>%

layer\_dense(units = 64, activation = "relu") %>%

layer\_dense(units = 64, activation = "relu") %>%

layer\_dense(units = 8, activation = "softmax")

model %>% compile(

optimizer = optimizer\_sgd(learning\_rate = 0.2),

loss = "categorical\_crossentropy",

metrics = "categorical\_accuracy"

)

}