Models and Hypotheses

```
library(conflicted)

library(kableExtra)
library(glue)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v ggplot2   3.5.1     v tibble    3.2.1
## v lubridate 1.9.3     v tidyr     1.3.1
## v purrr     1.0.2
```

```
conflict_prefer("filter", "dplyr")
```

```
## [conflicted] Will prefer dplyr::filter over any other package.
```

```
models <- tribble(
  ~Detectors, ~Stages, ~mAP, ~Vanishing, ~Mislabeling, ~Untargeted,
  "YOLOv3", 1, 33.7, "Object", "Class", "Class, Box, Object",
  "SSD", 1, 29.5, "Class", "Class", "Class, Box",
  "RetinaNet", 1, 36.5, "Class", "Class", "Class, Box",
  "Faster R-CNN", 2, 37.4, "RPN: Object; Det: Class", "Det: Class", "RPN: Object, Box; Det: Class, Box"
  "Cascade R-CNN", 2, 40.3, "RPN 1: Object; RPNs 2, 3 + Det: Class", "RPNs 2, 3: Class; Det: Class", "R
)

models
```

```
## # A tibble: 5 x 6
##   Detectors     Stages   mAP Vanishing                    Mislabeling Untargeted
##   <chr>          <dbl> <dbl> <chr>                        <chr>       <chr>
## 1 YOLOv3             1  33.7 Object                       Class       Class, Bo~
## 2 SSD                1  29.5 Class                        Class       Class, Box
## 3 RetinaNet          1  36.5 Class                        Class       Class, Box
## 4 Faster R-CNN       2  37.4 RPN: Object; Det: Class      Det: Class  RPN: Obje~
## 5 Cascade R-CNN      2  40.3 RPN 1: Object; RPNs 2, 3 + ~ RPNs 2, 3:~ RPN 1: Ob~
```

```
note_alpha <- function(txt, num, double_escape = FALSE) {
  glue("{txt}{footnote_marker_alphabet(num, double_escape = double_escape)}")
}

add_linebreak <- function(data) {
  data |> mutate(across(everything(), linebreak))
}

break_models <- function(col) {
  str_replace_all(col, coll(c("; " = ";\n", " R-CNN" = "\nR-CNN")))
}
```

```r
models <- models |>
  rename('{note_alpha("Stages", 1)}' := Stages, '{note_alpha("COCO mAP", 2)}' := mAP, '{note_alpha("Unta
  mutate(across(everything(), break_models)) |>
  add_linebreak()

models
```

```
## # A tibble: 5 x 6
##   Detectors  Stages\\textsuperscr~1 COCO mAP\\textsupers~2 Vanishing Mislabeling
##   <chr>      <chr>                  <chr>                  <chr>     <chr>
## 1 "YOLOv3"   1                      33.7                   "Object"  "Class"
## 2 "SSD"      1                      29.5                   "Class"   "Class"
## 3 "RetinaNe~ 1                      36.5                   "Class"   "Class"
## 4 "\\makece~ 2                      37.4                   "\\makec~ "Det: Clas~
## 5 "\\makece~ 2                      40.3                   "\\makec~ "\\makecel~
## # i abbreviated names: 1: `Stages\\textsuperscript{a}`,
## #   2: `COCO mAP\\textsuperscript{b}`
## # i 1 more variable: `Untargeted\\textsuperscript{d}` <chr>
```

```r
attack_header <- c(3, 3)
names(attack_header) <- c(" ", note_alpha("Attack Losses", 3, double_escape = TRUE))

models |>
  kbl(
    booktabs = TRUE,
    escape = FALSE,
    caption = "Detection models and attack losses. Full details are given in Appendix \\ref{app:mod_los
  ) |>
  kable_styling(
    position = "center",
    latex_options = "striped"
  ) |>
  add_header_above(c(" " = 3, "Targeted" = 2, " " = 1)) |>
  add_header_above(attack_header, escape = FALSE) |>
  footnote(
    alphabet = c(
      "In general, 1-stage detectors are quicker whereas 2-stage detectors are more accurate, though th
      "COCO mean Average Precision (mAP) is the primary metric on the COCO challenge.",
      "The training losses in detectors typically include the box regression loss (Box), the class loss
      "Untargeted attack targets all training losses in a model, i.e.\ the backpropagation loss."
    ),
    threeparttable = TRUE
  )
```

```r
sub_results <- function(col) {
  str_replace_all(col, coll(c(">" = "$>$")))
}

results <- tribble(
  ~hp, ~sig,
  "1-stage > 2-stage models (YOLOv3, SSD, RetinaNet > Faster R-CNN, Cascade R-CNN)", "YOLOv3, SSD > Ret
  "Targeted > Untargeted attack", "YOLOv3 only",
  "Vanishing > Mislabeling attack", "All",
  "Larger attack iterations", "All",
  "Less confident targets", "All",
```

Table 1: Detection models and attack losses. Full details are given in Appendix ??.

			Attack Losses[c]		
			Targeted		
Detectors	Stages[a]	COCO mAP[b]	Vanishing	Mislabeling	Untargeted[d]
YOLOv3	1	33.7	Object	Class	Class, Box, Object
SSD	1	29.5	Class	Class	Class, Box
RetinaNet	1	36.5	Class	Class	Class, Box
Faster R-CNN	2	37.4	RPN: Object; Det: Class	Det: Class	RPN: Object, Box; Det: Class, Box
Cascade R-CNN	2	40.3	RPN 1: Object; RPNs 2, 3 + Det: Class	RPNs 2, 3: Class; Det: Class	RPN 1: Object, Box; RPNs 2, 3 + Det: Class, Box

[a] In general, 1-stage detectors are quicker whereas 2-stage detectors are more accurate, though the 1-stage RetinaNet aims to be both quick and accurate. In a 2-stage detector, the input image passes through a Region Proposal Network (RPN) stage and a detection (Det) stage.

[b] COCO mean Average Precision (mAP) is the primary metric on the COCO challenge.

[c] The training losses in detectors typically include the box regression loss (Box), the class loss on the 80 COCO labels and/or the background class (Class), and the objectness loss on categorizing an image region as background or object (Object).

[d] Untargeted attack targets all training losses in a model, i.e. the backpropagation loss.

```
  "Larger perturb boxes", "All except mislabeling attack on Faster R-CNN",
  "Shorter perturb-target distance", "All",
  "Less accurate target COCO class", "Mixed",
  glue("Lower target {note_alpha('IOU', 2)} (untargeted attack only)"), "All",
  "More probable intended class (mislabeling attack only)", "Mixed",
)

results |>
  rename("Hypotheses (higher success for)" := hp, '{note_alpha("Accepted (across attacks and models)",
  mutate(across(everything(), sub_results)) |>
  add_linebreak() |>
  kbl(
    booktabs = TRUE,
    escape = FALSE,
    caption = "Hypothesis testing in the randomized attack (Sections \\ref{sec:rand_hp} and \\ref{sec:ra
  ) |>
  kable_styling(
    position = "center",
    latex_options = c(
      "striped",
      "hold_position"
    ),
  ) |>
  column_spec(2, width = "1.5in") |>
  column_spec(1, width = "1.25in") |>
  footnote(
    alphabet = c(
      "$p < .05$ for Wald z-test on logistic estimate",
      "intersection-over-union"
```

```
  ),
  escape = FALSE
)
```

Table 2: Hypothesis testing in the randomized attack (Sections **??** and **??**)

Hypotheses (higher success for)	Accepted (across attacks and models)[a]
1-stage > 2-stage models (YOLOv3, SSD, RetinaNet > Faster R-CNN, Cascade R-CNN)	YOLOv3, SSD > RetinaNet, Faster R-CNN, Cascade R-CNN in vanishing and mislabeling attacks (1-stage RetinaNet is as resilient as 2-stage models)
Targeted > Untargeted attack	YOLOv3 only
Vanishing > Mislabeling attack	All
Larger attack iterations	All
Less confident targets	All
Larger perturb boxes	All except mislabeling attack on Faster R-CNN
Shorter perturb-target distance	All
Less accurate target COCO class	Mixed
Lower target IOU[b] (untargeted attack only)	All
More probable intended class (mislabeling attack only)	Mixed

[a] $p < .05$ for Wald z-test on logistic estimate

[b] intersection-over-union