ManyLLMs: Analysis

Authors: XXX

2025-05-21

Contents

1 Overview

This script accompanies the analyses presented in the manuscript. Two versions are provided:

- The `.Rmd` }le can be opened and edited in RStudio.
- A rendered PDF is available for reference, generated by knitting this }le.

We provide full R code to ensure the analyses are transparent and reproducible.
Where necessary, code is explained or unpacked in-text. This document includes:

1. Visualizations of key variable distributions (used in main text and supplementary materials)
2. Justi}cations for major analysis decisions

Last updated: 21 May 2025

First, we recoded factor levels and renamed variables in the dataset to improve readability and streamline the subsequent analysis..

1.1 Dataset explaination

Here we summarize the important Variables in our dataset.

Our dataset captures each LLM's response to a moral dilemma in the `opinion` column— a 1–7 utilitarian score where higher scores indicate a stronger willingness to endorse the "greater good" at the expense of a moral rule.

The variable `model` identi}es the type of LLMs which produced this response (e.g.,GPT4.1,Llama3, etc) The }eld `item` (or `example_index`) numbers the scenario, and `rep` indicates the repetition for each scenario (1, 2, 3..).

The }eld `Group_Size` codes group size (`1` = solo, `2` = pair, `3` = triad).

The variable `type` classi}es the moral dilemma (e.g., "Personal", "Killing–Util") within the }eld `dataset` which names the source questionnaire (e.g., `greene`, `oxfor`, `korner_cni`) which are our measures. This is similar to the variable `measurement`.

Finally, the variable `Group` denotes the condition: `Solo` for the single agents (baseline of the model) and `Group` for the }nal group-re~ection consensus. The variable `step` is the same, but numerical: Solo (`step -1`) vs Group(`step -1`). Note that `Group` indicates both pairs and triads. This is our main experimental manipulation.

We base our analysis on these sets measures to capture utilitarian boost in groups of multi-agent LLM settings. Therefore, in our analysis, we always compare `Group` vs. `Solo` condition. You can see a summary of the variables in Table 1.

```
glimpse(combined_dataset)
```

Table 1: Table: Variable De}nitions

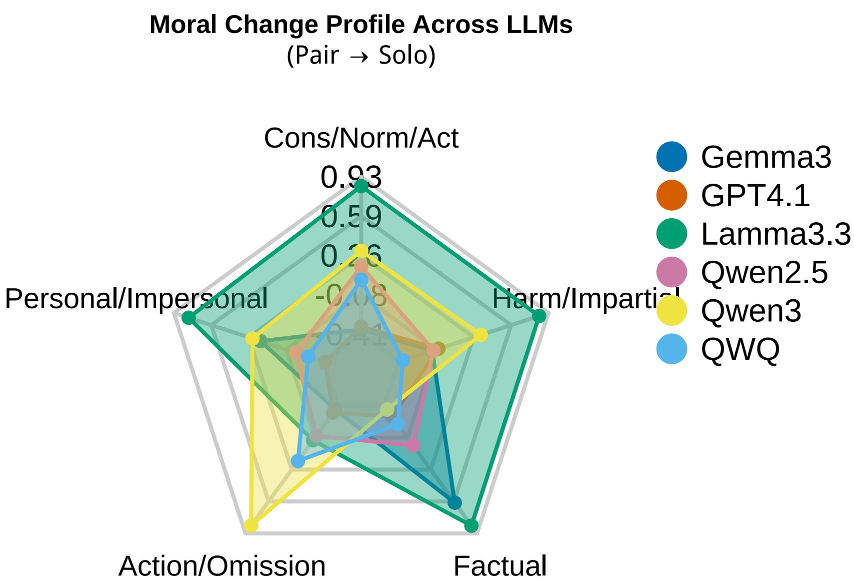
Variable	Description
model	LLM that generated the response (e.g., gpt-4.1, llama3.3)
item	ID of the dilemma scenario - a.k.a expample_ index
rep	Repetition index for each scenario (e.g.,1, 2, 3)
group	Solo (baseline) and Group (consensus)
ob/ group_ size	Group size: n = solo, nn = pair, nnn = triad
type	Moral category of the scenario (e.g., Personal, Impersonal, Killing–Util)
dataset	Measurement (e.g., greene, oxford, korner, cni)
opinion	Utilitarian Score: LLMs' ratings on a 1–7 Likert scale (higher = more utilitarian)

1.2 Moral Chage Across LLMs

Together our varriables let us track, for each model × item × phase × repetiion, how the LLM's moral judgment shifts from Solo to Group.

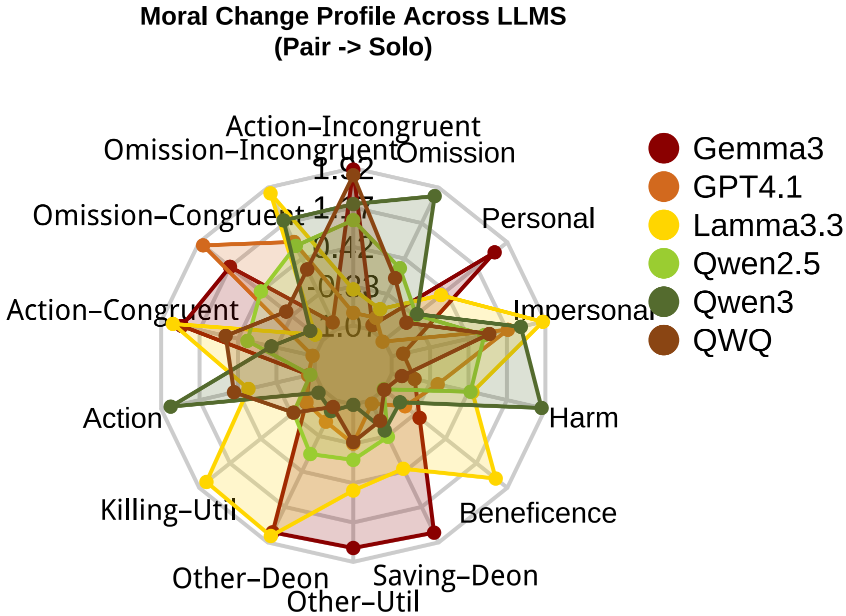
1.2.1 Pair vs Solo across all the diﬀerent measurements

Now We plot Group vs Solo across all the diﬀerent measurements using an Radar plot.



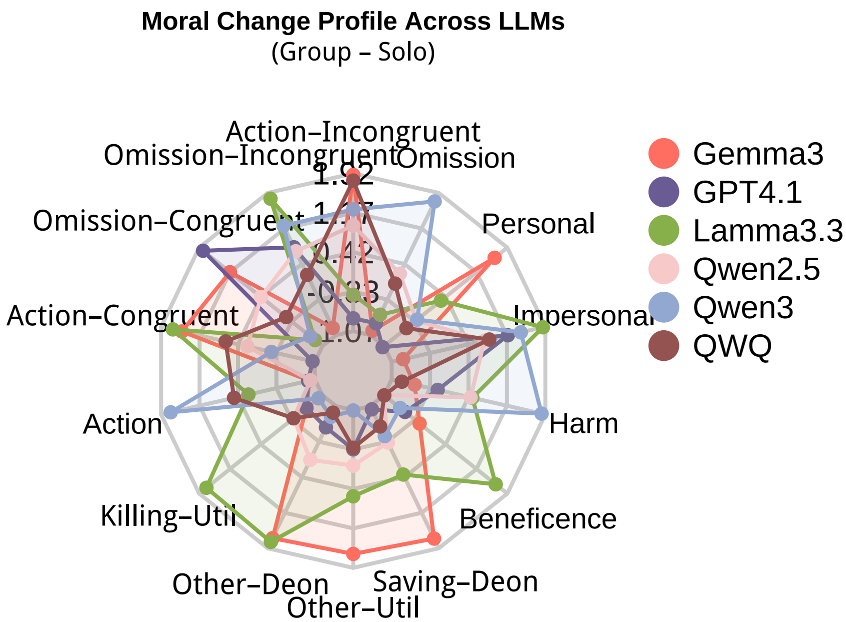
Pair vs Solo across all the diﬀerent types

This shows moral change pro}les across diﬀerent models for diﬀerent types The diﬀerence is the `Pairs - Solo`.



Group vs Solo across all the diﬀerent types

This shows moral change pro}les across diﬀerent models. The diﬀerence is the `Groups (pairs and triads) - Solo`.



We see the pro}le of moral change is diﬀerent across models and types. Next we See each measurement in more depth.

2 Utilitarain Moral Dilemmas

Our primary reference is the classic set of moral utilitarian dilemmas ('dataset == "greene").

```r
# Recreate the Greene subset for this document
greene_df <- combined_dataset %>%
  filter(dataset == "greene") %>%
  droplevels()
```

Preparing Data for Ordinal Modeling: We begin by extracting only the Greene subset and converting our response variable to an ordered factor:

```r
# Prepare dataset
reflection_moral <- greene_df
reflection_moral$opinion <- as.ordered(reflection_moral$opinion)
```

2.1 Models

The ordinal package[1] provides functions for fitting cumulative link models (`clm()`) and cumulative link mixed models (`clmm()`) to ordinal response data. It supports both fixed-effects formulae and random-effects structures, making it ideal for analyzing Likert-type outcomes with clustered or repeated measures.

Next, we fit three candidate cumulative link mixed models to determine the optimal random-effects structure. Each model includes the fixed effect of `Group` (factor type of step) but varies in its random terms:

Note: The variable `rep` denotes repetition—each scenario is repeated multiple times to ensure consistency and robustness of the judgments.

```r
model1 <- clmm(
  opinion ~ Group + (1 | item),
  data  = reflection_moral,
  Hess  = TRUE
)
```

1. Model 1: random intercept for each `item`
   This baseline model uses `(1 | item)` to allow every dilemma scenario to have its own starting point on the 1–7 scale. It assumes that the effect of `Group` (single vs. group) is constant across all items, but accounts for inherent differences in how "utilitarian" each scenario tends to be.

```r
model2 <- clmm(
  opinion ~ Group + (rep | item),
  data  = reflection_moral,
  Hess  = TRUE
)
```

2. Model 2: random slope of repetitions `rep` within each `item`
   In addition to item intercepts, this model lets each scenario exhibit its own pattern across repeated presentations. Some items may elicit more consistent ratings across repeats, while others show greater variability, capturing item-specific response stability.

```r
model3 <- clmm(
  opinion ~ Group + (1 | item)+ (1 | rep),
  data  = reflection_moral,
  Hess  = TRUE
)
```

[1] Christensen, R. H. B. (2019). ordinal: Regression Models for Ordinal Data. R package version 2019.12-10. https://CRAN.R-project.org/package=ordinal

3. Model 3: crossed random intercepts for `item` and `rep`
   This model builds on the previous ones by adding a separate random intercept for each repetitions (`rep`) in addition to the scenario-specific intercepts for each `item`. This accounts for two sources of baseline variation:

   - Item-level: some dilemmas are judged more utilitarian or deontological on average.
   - Repetition-level: certain runs (e.g., the first, second, third presentation) may systematically differ.

2.2 Models Comparison and Selection

Model Comparison: with Likelihood-Ratio Test: We compare the three fitted `clmm` models using `anova()` to perform a likelihood-ratio test. This assesses whether each increase in complexity of the random-effects structure leads to a statistically significant improvement in model fit.

```
anova_res <- anova(model1, model2, model3)
```

Table 2: Table: Likelihood-Ratio Test for Random-Effects Structures

Model	AIC	Chi-square	df	p-value
Model 1	38759.18	NA	NA	NA
Model 3	38761.18	0.00	1	0.9689
Model 2	38652.66	110.52	1	0.0000

2.2.1 Model Selection

Model 2 is the preferred model. The likelihood-ratio test comparing Model 2 to Model 1 yields p = 0.969, indicating a significant improvement in fit. Additionally, Model 2 reduces the AIC by 106.53 and the BIC by 91.04 relative to Model 1. These metrics together demonstrate that allowing item-specific repetition effects (Model 2) provides a more parsimonious and better-fitting model.

2.3 Utilitarian Boost

Model 2 was selected as the optimal random-effects structure. Below we summarize its fixed-effect estimates and then report pairwise comparisons of reflection phases on the probability scale.

We now report the fixed-effect estimates and pairwise comparisons for Model 2 in Table 2.

Table 3: Table: Fixed-Effect Estimates for Model 2

Term	Estimate	Std. Error	z value	p value
1|2	-3.02218	0.39612	-7.62951	0.00000
2|3	-1.04930	0.39530	-2.65446	0.00794
3|4	-0.11148	0.39520	-0.28208	0.77788
4|5	0.45193	0.39525	1.14342	0.25286
5|6	1.48428	0.39546	3.75331	0.00017
6|7	3.68671	0.39665	9.29454	0.00000
GroupSolo	-0.31754	0.04561	-6.96163	0.00000

2.4 Personal vs. Impersonal

Our primary reference is the classic set of moral sacri}cial dilemmas (`dataset == "greene"`), which we analyze across two key dimensions:
- `type == "Personal"` vs. `type == "Impersonal"`

Using our winning random-effects structure, we now test whether the effect of Group (`Group`) differs across scenario types (`type`). We }t a cumulative link mixed model with an interaction between `Group` and `type`, retaining a random intercept for each dilemma item (`item`).

```r
#Fit cumulative link mixed model (CLMM)
model_clmm <- clmm(
  opinion ~ Group * type +
    (1| item),
  data = reflection_moral,
  Hess = TRUE
)

# Summarize the fitted model
summary(model_clmm)
```

```
## Cumulative Link Mixed Model fitted with the Laplace approximation
##
## formula: opinion ~ Group * type + (1 | item)
## data:    reflection_moral
##
##  link  threshold nobs  logLik    AIC      niter       max.grad cond.H
##  logit flexible  17049 -19326.86 38673.73 1049(12482) 6.67e-04 3.3e+04
##
## Random effects:
##  Groups Name        Variance Std.Dev.
##  item   (Intercept) 11.87    3.446
## Number of groups:  item 44
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## GroupSolo                 0.10994    0.05946   1.849   0.0645 .
## typePersonal             -1.90090    1.05572  -1.801   0.0718 .
## GroupSolo:typePersonal   -0.71708    0.07816  -9.175   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Threshold coefficients:
##     Estimate Std. Error z value
## 1|2  -3.7103     0.8235  -4.505
## 2|3  -1.7499     0.8232  -2.126
## 3|4  -0.8099     0.8232  -0.984
## 4|5  -0.2456     0.8231  -0.298
## 5|6   0.7877     0.8231   0.957
## 6|7   2.9890     0.8232   3.631
```

We then extract pairwise estimated marginal means on the probability scale and present the contrast results in a PDF-friendly table.

```r
# 4. Obtain pairwise estimated marginal means (on the probability scale)
emms <- emmeans(
  model_clmm,
  pairwise ~ Group * type ,
  type = "response"
)
# 5. View contrast summaries
summary(emms$contrasts)
```

```
##  contrast                                  estimate     SE  df z.ratio p.value
##  Group Impersonal - Solo Impersonal          -0.110 0.0595 Inf  -1.849  0.2505
##  Group Impersonal - Group Personal            1.901 1.0600 Inf   1.801  0.2730
##  Group Impersonal - Solo Personal             2.508 1.0600 Inf   2.377  0.0816
##  Solo Impersonal - Group Personal             2.011 1.0500 Inf   1.907  0.2251
##  Solo Impersonal - Solo Personal              2.618 1.0500 Inf   2.484  0.0625
##  Group Personal - Solo Personal               0.607 0.0507 Inf  11.969  <.0001
##
## P value adjustment: tukey method for comparing a family of 4 estimates
```

P value adjustment: tukey method for comparing a family of 4 estimates

Wihtout rep: (Group-1 Impersonal) - Group7 Impersonal 0.111 0.0594 Inf 1.874 0.2393 (Group-1 Impersonal) - (Group-1 Personal) 2.616 0.9380 Inf 2.789 0.0271 (Group-1 Impersonal) - Group7 Personal 2.021 0.9390 Inf 2.154 0.1364 Group7 Impersonal - (Group-1 Personal) 2.504 0.9390 Inf 2.667 0.0383 Group7 Impersonal - Group7 Personal 1.910 0.9400 Inf 2.033 0.1760 (Group-1 Personal) - Group7 Personal -0.594 0.0507 Inf -11.728 < .0001

Table 4: Contrasts Within Type (Impersonal and Personal)

contrast	estimate	SE	z.ratio	p.value
Group Impersonal - Solo Impersonal	-0.10994	0.05946	-1.84893	0.25046
Group Personal - Solo Personal	0.60714	0.05072	11.96920	0.00000

We see that **Group** opinions are signi}cantly higher than Solo opinions across all LLMs for **Personal** dilemmas.

2.5 Pairs and Triads: group size e ect.

In our previous models, including a random slope for **repetition** (`rep`) caused convergence failures, so we removed `rep` from the random structure and now focus on how **group size** (`ob`) a ects moral judgments.

Because only the **Personal** dilemma type showed a signi}cant interaction with re~ection phase, we restrict our analysis to **Personal** scenarios. We then }t a cumulative link mixed model with:

- **Fixed e cts:** interaction of `Group` (“Solo” vs. “Group”) and `ob` (n, nn, nnn)
- **Random intercepts:** for each scenario item (`item`)

```r
# Fit cumulative link mixed model (CLMM)
model_clmm <- clmm(
  opinion ~ Group * groupsize  +
```