Visualizing predictions

February 15, 2021

1 Introduction

In this task, we describe various R machineries for displaying predictions in both simple and complex (involving interaction terms) generalized linear models. We first consider existing approaches for constructing model predictions and then describe how our proposed approach is different from the existing ones. When visually presented, predictions provide a unified and intuitive way of describing relationships from a fitted model, especially complex models involving interaction terms or some kind transformations on the dependent variables whose estimates are usually, but not always, a subject to less clarity of interpretation.

A related term to *marginal* predictions is the *conditional* predictions — it is not clear whether conditional or marginal predictions is more appropriate for a particular research problem [Muff et al., 2016]. However, we attempt to make the distinction between the two (see glossary).

By default, commonly used R packages for constructing predictions do not remove (*zero-out*) the uncertainties resulting from the *non-focal* predictors when computing predictions. A non-trivial way to achieve this in some of these packages is to provide a user defined variance-covariance matrix with the covariances of *non-focal* terms set to 0 – *zeroing-out* variance-covariance matrix. This only works when the predictors are centered prior to model fitting, in case of numerical predictors, and even much complicated when the predictors are categorical. We propose a method independent of model scaling to provide more robust *marginal* predictions.

We can derive various quantities from a fitted regression model. The first and most obvious, is the model coefficient estimates. Others include predicted values of the outcome variable — 1) predictions at a particular; and 2) mean or median value of the predictors. The first case simply involves evaluation of the fitted model function, say $\hat{f}(X = x)$, at some particular value of x. The second case, chooses the values of the predictor based on its distributional properties.

Most importantly, from these predicted values, we can also generate second class quantities of interest — *marginal* predictions, which describes the change in the predicted value of the dependent variable after changing one independent variable – either a discrete change in the categorical variable(s) or an instantaneous change in continuous variables, while all other variable are held at specified values.

Suppose we are interested in the *marginal* predictions of a particular predictor (hence forth referred as *focal* predictor otherwise *non-focal*), x_f, in the set of predictors. To keep it simple, assume that the model has no interaction terms. Then the idea is to fix the values of *non-focal* predictor(s) at some typical values – typically determined by averaging in some meaningful way, for example, arithmetic mean and average over the levels of the factors of *non-focal* continuous and categorical predictors, respectively. An alternative to *averaging* is *anchoring* which involves picking a fixed value of the predictor. One way to achieve this is using *model matrix*, $\mathbf{X}$, i.e., *anchoring* or *averaging* the columns of *non-focal* terms in $\mathbf{X}$.

1.1 Simple linear models

Consider a simple linear model with linear predictor $\eta = \mathbf{X}\beta$ and let $g(\mu) = \eta$ be an identity link function (in the case of simple linear model), where μ is the predicted (expected) value of outcome variable y. Let $\hat{\beta}$ be the estimate of β, together with the estimated covariance matrix $V(\hat{\beta})$ of $\hat{\beta}$. Let the entries of $\mathbf{X}^*$ include all *non-focal* and *focal* predictors. The model matrix, $\mathbf{X}^*$, inherits most of its key properties, for example transformation (e.g., scaling) on the predictors and interactions from the model matrix, $\mathbf{X}$. Then the predicted values $\hat{\eta}^* = \mathbf{X}^*\hat{\beta}$ represents the *marginal* predictions of the focal predictor. Alternatively, we can transform these predictions to response scale using $g^{-1}(\hat{\eta}^*)$.

Further, we can compute the standard errors (SEs) associated with these predictions, $\hat{\eta}^*$, for constructing confidence intervals, as the $\mathrm{sqrt}(\mathrm{diag}(\mathbf{X}^* V(\hat{\beta})\mathbf{X}^{*T}))$.

Our proposed method goes one step further of removing the *uncertainty* as a result of *non-focal* predictors are removed. This can be achieved in two ways:

- using variance-covariance matrix, $V(\hat{\beta})$
- using centered model matrix, $\mathbf{X}^*$

1.1.1 Variance-covariance

The computation of $\hat{\eta}^*$ remains the same as described above. However, to compute SEs $V(\hat{\beta})$ is modified by *zeroing-out* (the variance-covariance of all non-focal predictors are assigned zero) entries of *non-focal* terms in $V(\hat{\beta})$. This approach requires *centering* of the predictors in the model matrix, X. In other words, the fitted model should have *centered* predictors.

1.1.2 Centered model matrix

Suppose the *non-focal* entries in X^* are computed by some kind of averaging or anchoring. Consider centered X^*, $X^*_c = (X^* - \bar{X}^*)$. It follows that the *non-focal* entries in X^*_c are all zero. As a result, the SEs are computed as $\mathrm{sqrt}(\mathrm{diag}(\mathbf{X}^*_c V(\hat{\beta})\mathbf{X}^{*T}_c))$, which actually *zeros-out*. More generally, *centering*, $\mathbf{X}^*_c = \mathbf{X}^* - k$ (for example $k = E(\mathbf{X}^*)$) impacts on the estimated value of the intercept and its associated variance. However, the slopes are not affect by this. The implication of this that since *non-focal* terms in $\mathbf{X}^*_c$ are zero, it doesn't matter what their corresponding values are in the variance-covariance matrix. Hence, we can compute *marginal* predictions from non-centered predictors (in other words, fitted models with predictors in their natural scales).

1.2 GLMs

1.3 LMEs

2 Available packages

The following R packages **effects**, **emmeans** and **margins** implement various schemes for constructing *marginal* predictions. However, currently, their ability to compute *marginal* predictions with zeroed-out non-focal uncertanities is limited to the use of variance-covariance approach which requires the fitted model to be centered. We propose **varpred** to overcome this limitation.

3 TODOs

1. Models with interactions
2. GLMs
3. LMEs

4 No interactions

4.1 Simulation

Consider a simple no-interaction terms simulation:

- $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$ s.t $\epsilon \sim N(0, 1)$ and $\{\beta_0 = -5, \beta_1 = 0.1, \beta_2 = -0.6\}$
  - $x_1 \sim unif(1, 9)$
  - x_2 – two level categorical variable

```
       x1u x2u          y       x1c       x1s       x1sd
1 3.977587   A -0.4524966 -1.222599 0.6940038 -0.5134124
2 1.350599   A  0.1605887 -3.849588 0.2356505 -1.6165773
3 6.677472   A  1.5456482  1.477286 1.1650760  0.6203644
4 6.261523   A -0.4707503  1.061337 1.0925018  0.4456928
5 2.998846   A  1.3489423 -2.201340 0.5232344 -0.9244202
6 3.400439   A -0.4111428 -1.799747 0.5933038 -0.7557773
```

4.2 Model fitting

No scaling or centering covariates

```
> df_temp <- drop(df)         # margins doesn't work with transform
>                            # but we need df to extract attributes :(
> lm_u <- lm(y ~ x1u + x2u, data = df_temp)
```

Centered covariates $(x - \bar{x})$ model

```
> lm_c <- lm(y ~ x1c + x2u, data = df_temp)
```

Scaled covariates (x/sdx) model

```
> lm_s <- lm(y ~ x1s + x2u, data = df_temp)
```

Both scaled and centered covariates

```
> lm_sd <- lm(y ~ x1sd + x2u, data = df_temp)
```

Coefficients estimates

```
> coef_est <- modsummary(fun=coef)
> print(coef_est)
```

```
              lm_c   lm_s  lm_sd   lm_u
(Intercept)  0.808  0.229  0.808  0.229
x1c          0.111  0.638  0.265  0.111
x2uB        -0.535 -0.535 -0.535 -0.535
```

These four model are equivalent (they all have equal loglikehoods).

```
> ll_est <- modsummary(fun=logLik)
> print(ll_est)
```

```
        lm_c    lm_s   lm_sd    lm_u
[1,] -272.27 -272.27 -272.27 -272.27
```

Variance covariance matrix. It is important to note that the slope variance for unscaled and centered models are the same. This makes it easier to center predictions from uncentered models.

```
> vcov_est <- modsummary(fun=function(x)vcov(x), simplify=TRUE
+         , combine="rbind", match_colnames = names(coef(lm_u)))
> print(vcov_est)
```

```
                  (Intercept)    x1u   x2uB model
lm_c.(Intercept)        0.009  0.000 -0.009  lm_c
lm_c.x1c                0.000  0.001  0.000  lm_c
lm_c.x2uB              -0.009  0.000  0.018  lm_c
lm_s.(Intercept)        0.031 -0.024 -0.009  lm_s
lm_s.x1s               -0.024  0.026  0.000  lm_s
```

```
lm_s.x2uB              -0.009  0.000  0.018  lm_s
lm_sd.(Intercept)       0.009  0.000 -0.009 lm_sd
lm_sd.x1sd              0.000  0.005  0.000 lm_sd
lm_sd.x2uB             -0.009  0.000  0.018 lm_sd
lm_u.(Intercept)        0.031 -0.004 -0.009  lm_u
lm_u.x1u               -0.004  0.001  0.000  lm_u
lm_u.x2uB              -0.009  0.000  0.018  lm_u
```

4.3 All uncertainties included

4.3.1 Continuous predictor

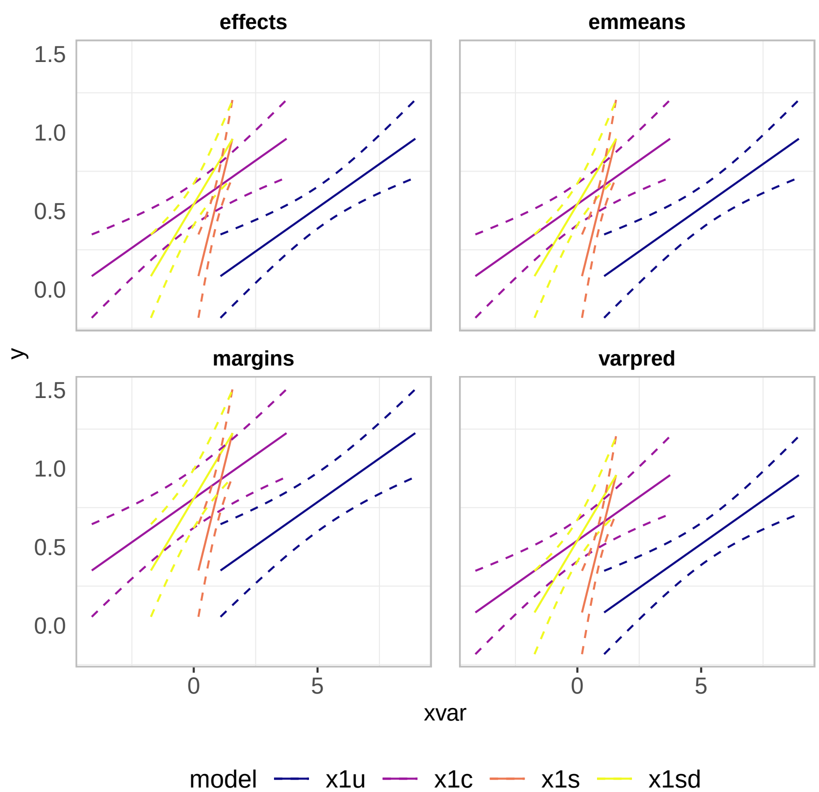


Figure 1: Predictions on the variable-specific values.

Figure 1, displays the simple marginal predictions for each of the four models described above, i.e., variable-specific values (original (u), mean centered (c), divided by standard deviation (s), and both mean centered and scaled (sd)). The uncertainties associated with these predictions includes those for non-focal predictors too. The predictions looks different but they are actually similar when plotted on the same scale, see Figure 2.

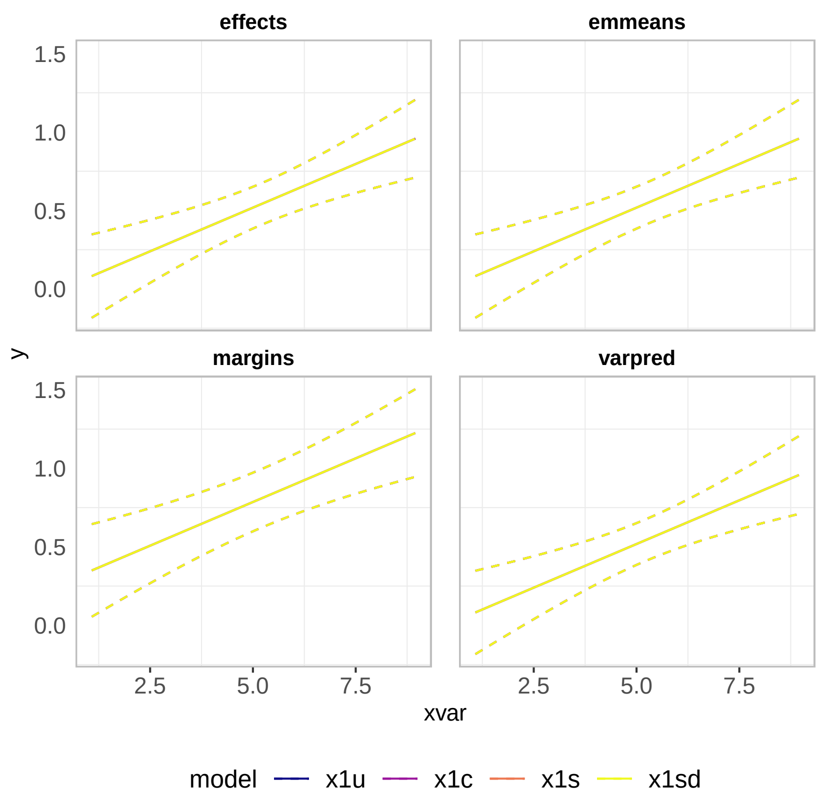


Figure 2: Back-transformed conditional predictions. Remember the model predictors were in different scales – this implies that if we know the scaling or centering parameter, we can always back-transform model predictions to the original (unscaled) form.

4.3.2 Categorical predictors

In this case, it doesn't matter which of the four models we choose since they all give the similar predictions.

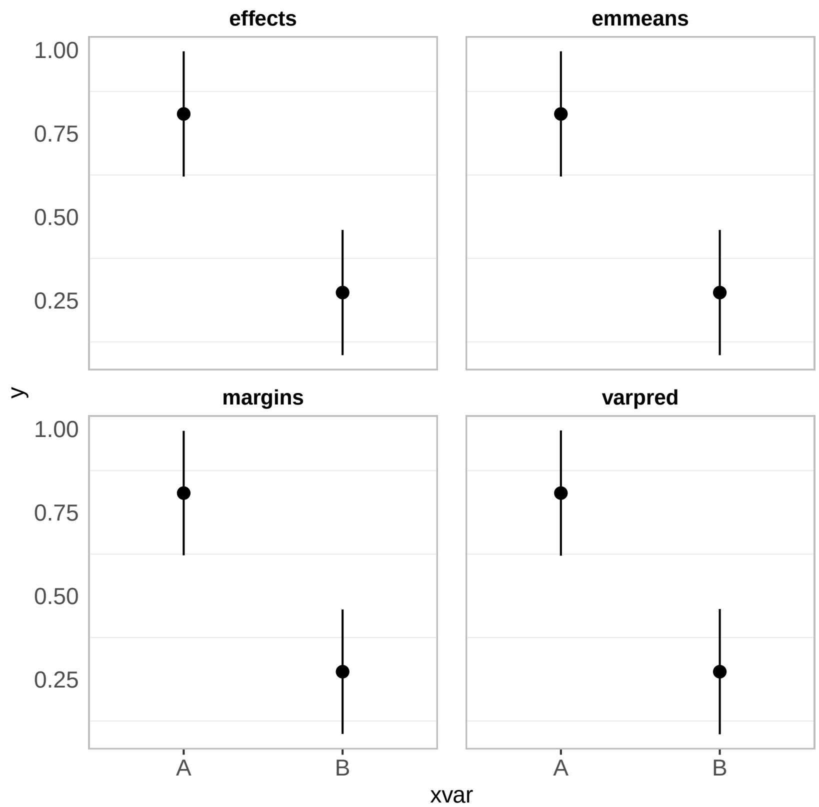


Figure 3: Conditional predictions.

4.4 Zeroed-out uncertainties

We can remove uncertainties associated with non-focal predictors by *zeroing-out* the variance of non-focal terms of the variance-covariance matrix (possible in all four methods) but this requires that continuous predictors are centered or appropriate contrast is applied in the case of categorical predictors before model fitting. In **varpred** marginal predictions are computed by centering the model matrix (when computing SEs), whic does not require centered predictors and is independent of contrast specification.

4.4.1 zeroed-out covariance matrix

The modification in each of the methods is to either specify *zeroed-out* covariance matrix or a function to compute the variance-covariance matrix as one of the inputs. In particular, in **varpred**, **emmeans** and **effects** it is specified through **vcov.** and in **margins** through **vcov** argument. However, this seems not to work currently in **cplot** which computes the predictions in **margins**.
The **zero_vcov** function in **jdeffects** *zeros-out* variance-covariances of non-focal terms of the predictor in question. For example:

- full variance-covariance matrix

```
            (Intercept)           x1u          x2uB
(Intercept)  0.030729586 -4.169417e-03 -9.047842e-03
x1u         -0.004169417  8.017823e-04 -3.588329e-19
x2uB        -0.009047842 -3.588329e-19  1.809568e-02
```

- *zeroed-out* variance-covariance matrix with `x1u` as the focal variable

```
           (Intercept)          x1u x2uB
(Intercept)          0 0.0000000000    0
x1u                  0 0.0008017823    0
x2uB                 0 0.0000000000    0
```

For the continuous predictors, we first compute predictions on predictor-specific scales (Figure 4) and then back-transform the predictions to original scale (unscale – uncenter and/or unscale). To clearly distinguish between what happens when apply various scaling schemes, we separately do the plots (see Figure 5).

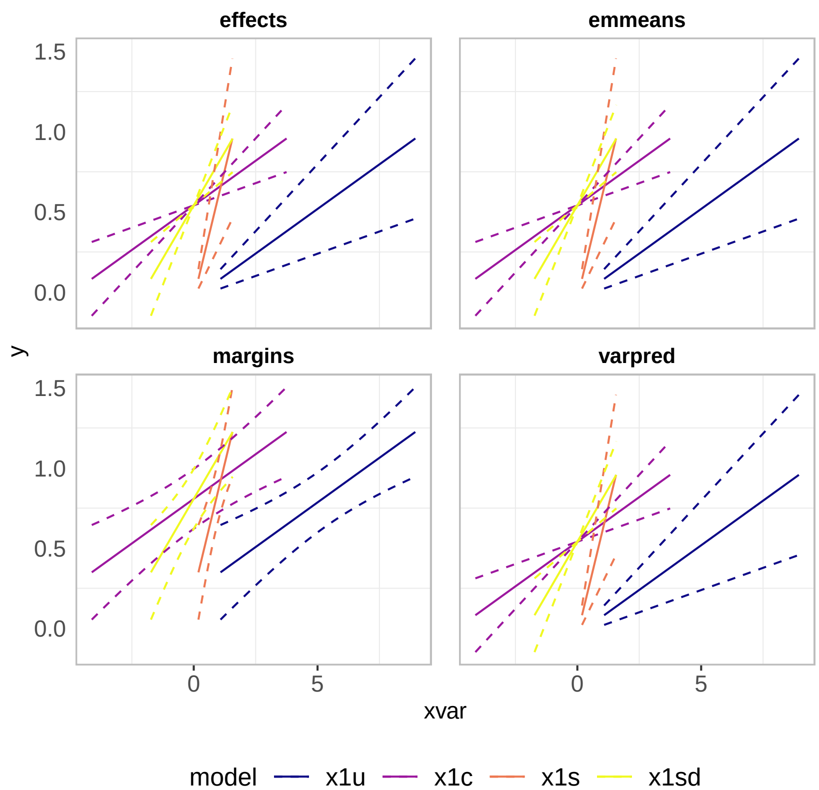
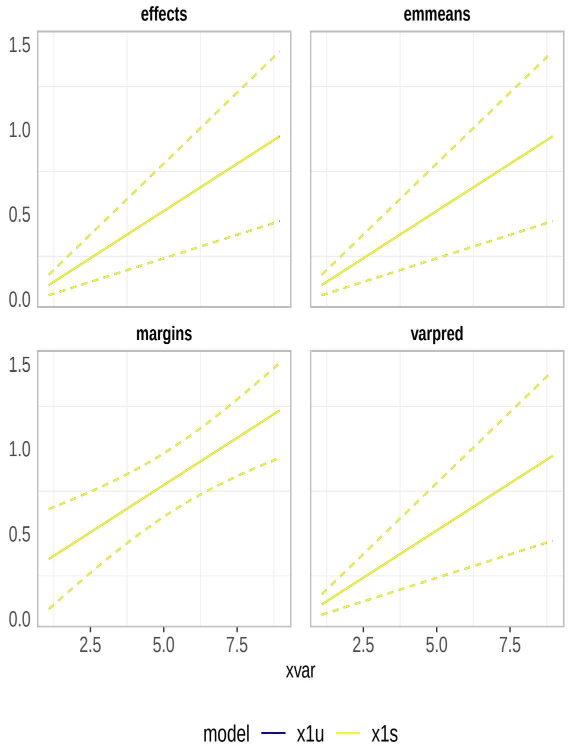


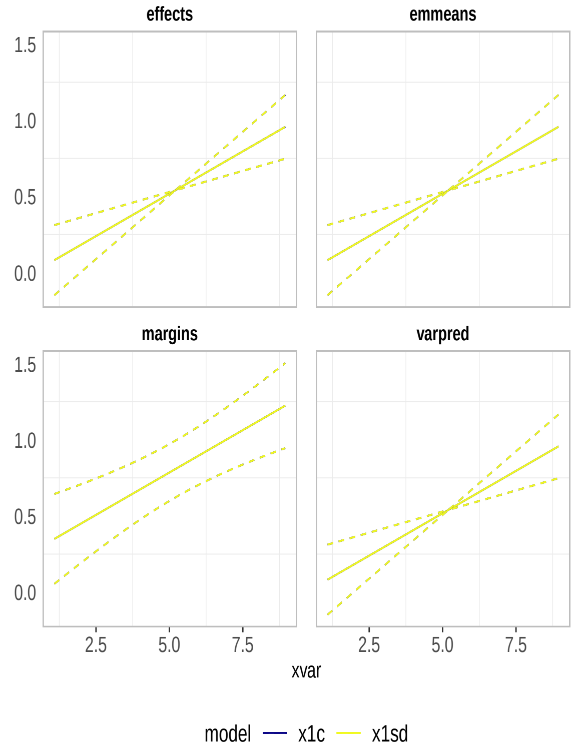
Figure 4: Marginal predictions with the zeroed-out covariances on the variable-specific scale.

4.4.2 Centered model matrix

Marginal predictions in **effects** and **emmeans** require models fitted with transformed predictors. However, in **varpred** we can use model fitted using predictors on their original scale and estimate marginal predictions. In our example, we use the unscaled model `lm_u` and simply set `isolate=TRUE`. The predictions (see Figure 6) are similar to Figure 5 b (except **margins**).



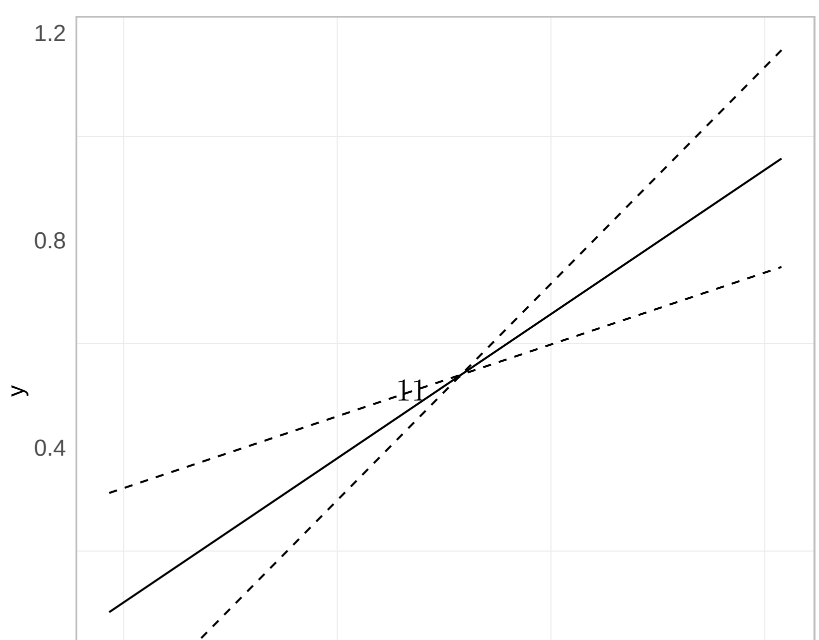
(a) Either unscaled or devided by the *sd*.



(b) Centered or centered and/or devided by *sd*.

Figure 5: Back-transformed marginal predictions.

```
> ## Centered predictions from unscaled model
> vpred_c <- varpred(lm_u, focal = "x1u", isolate = TRUE)
> plot(vpred_c)
```



4.4.3 Categorical predictors

Back to categorical predictor, `x2u`. The default contrast compares the other categories with no-variance reference category (base) and as a result, we can not compute the SEs using *zeroed-out* variance-covariance matrix unless appropriate contrast (`"contr.sum"`) was used during model specification. Model matrix centering provides a way to overcome this and computes centered independent of the contrasts. Currently, only **varpred** provide this functionality.

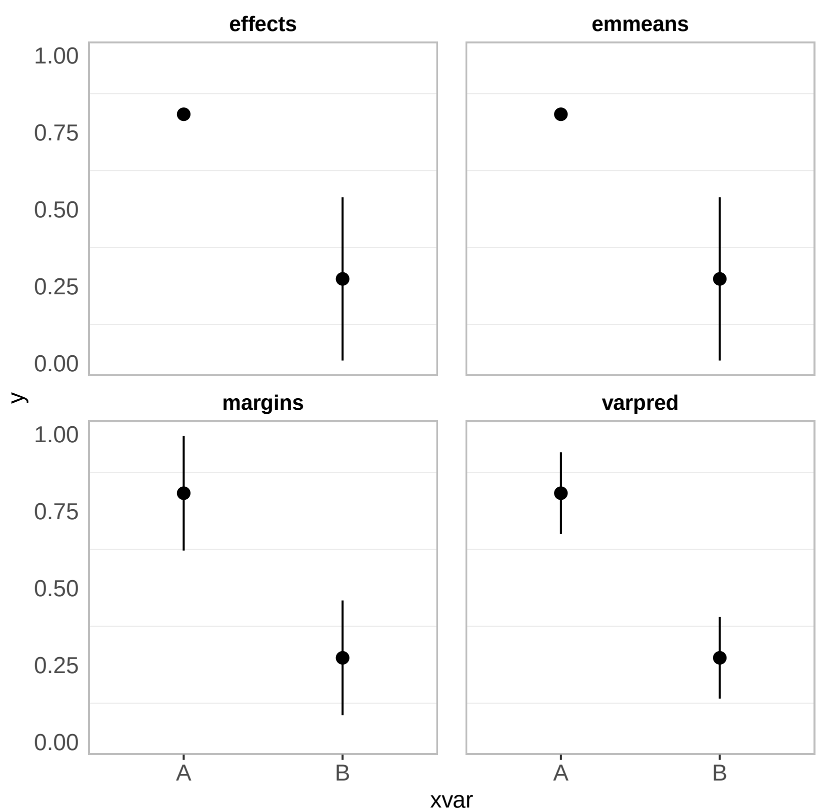


Figure 7: Conditional predictions.

4.4.4 Comparison with base R predict

Compare prediction estimates from **base R**, **emmeans** and **varpred**

```
> ## Base R vs emmeans
> all.equal(base, emm)

[1] TRUE

> ## Base R vs varpred
> all.equal(base, jd)

[1] TRUE

> ## Implied: emmeans vs varpred
```

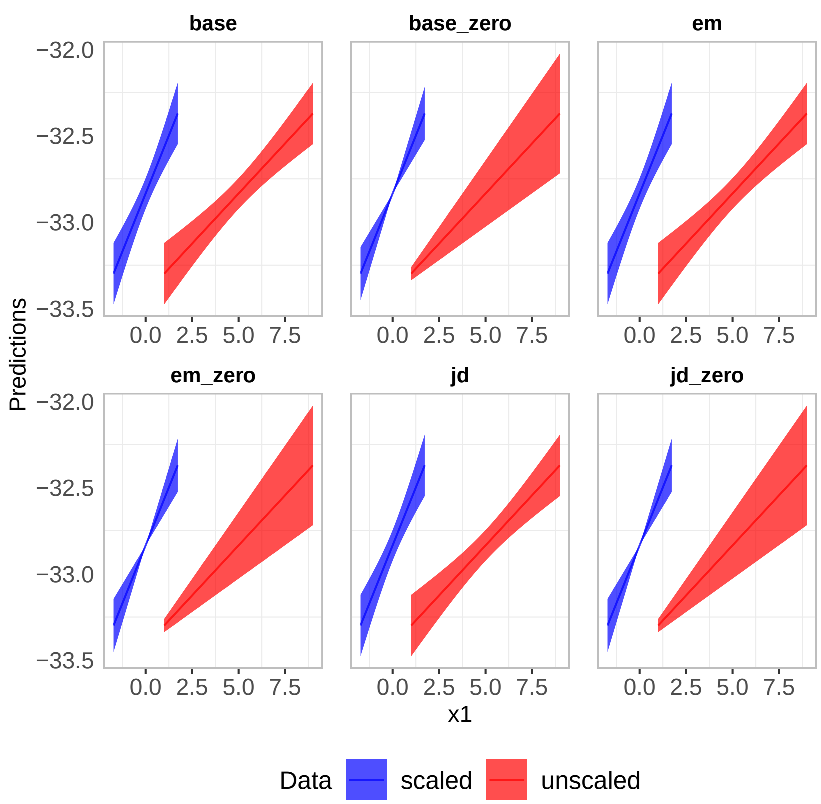


Figure 8: Conditional predictions.

5 Unbalanced data

Consider a simple no-interaction terms simulation:

- $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \epsilon$ s.t $\epsilon \sim N(0,1)$ and $\{\beta_0 = -5, \beta_1 = 0.1, \beta_2 = \{-0.6, 0.5\}\}$
  - $x_1 \sim unif(1,9)$
  - x_2 – three level categorical variable s.t A, B and C are 10%, 30% and 60%, respectively.

```
    A     B     C 
0.088 0.294 0.618 
```

```
       x1u x2u           y x2c       x1s       x1sd
1 3.977587   C -0.87459943   C 0.7156429 -0.4637524
2 1.350599   B -1.62801173   B 0.2429981 -1.6002413
3 6.677472   C  0.37120408   C 1.2014031  0.7042730
4 6.261523   C  0.05218589   C 1.1265660  0.5243250
5 2.998846   B -0.82628950   B 0.5395489 -0.8871759
6 3.400439   B -0.50022891   B 0.6118030 -0.7134386
```

We fit two models – the model without contrast and the one with sum-to-contrast (‘`contr.sum`’):

```
> ## No contrast model
> lm_no_contr <- lm(y ~ x1u + x2u, data = df)
> print(lm_no_contr)

Call:
lm(formula = y ~ x1u + x2u, data = df)

Coefficients:
(Intercept)          x1u         x2uB         x2uC  
    -0.4074       0.1133      -0.6541       0.2562  

> ## With contrast
> contrasts(df$x2c) <- "contr.sum"
> lm_contr <- lm(y ~ x1u + x2c, data = df)
> print(lm_contr)
```

```
Call:
lm(formula = y ~ x1u + x2c, data = df)

Coefficients:
(Intercept)          x1u         x2c1         x2c2
    -0.5400       0.1133       0.1326      -0.5215
```

Figure 9 compares variable effects for the three packages when the variance of the non-focal predictors are included.

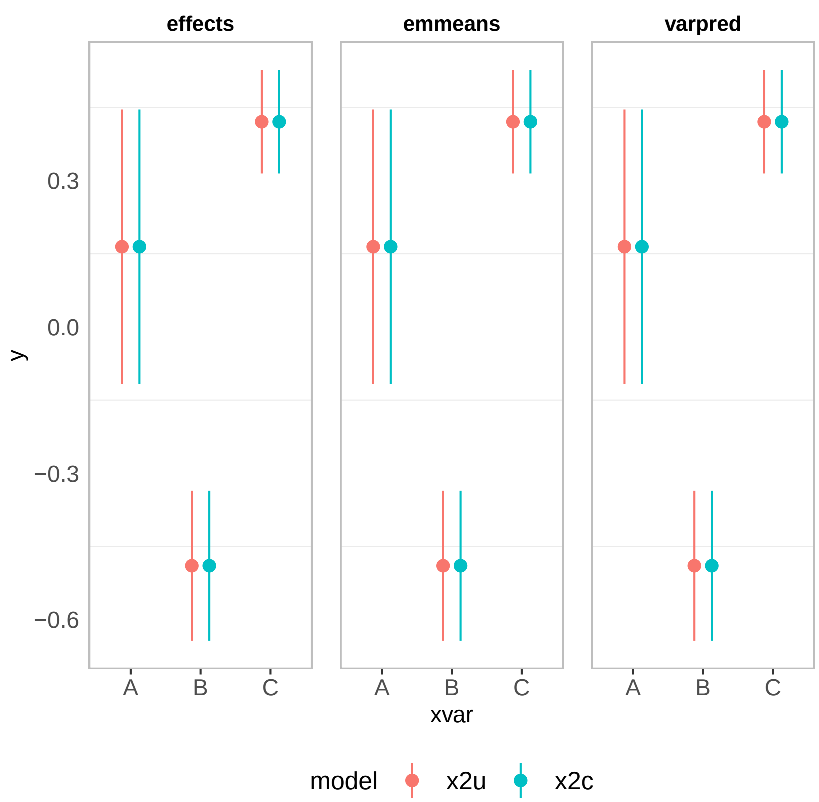


Figure 9: Variable effects with non-focal predictors' uncertainties included. The legend compares the model with non contrast (`x2u`) to the one with sum to zero contrast (`x2c`).

5.1 Zeroed-out uncertainties

As explained in the previous section, we again eliminate the uncertainties due to non-focal predictors.

5.1.1 zeroed-out covariance matrix

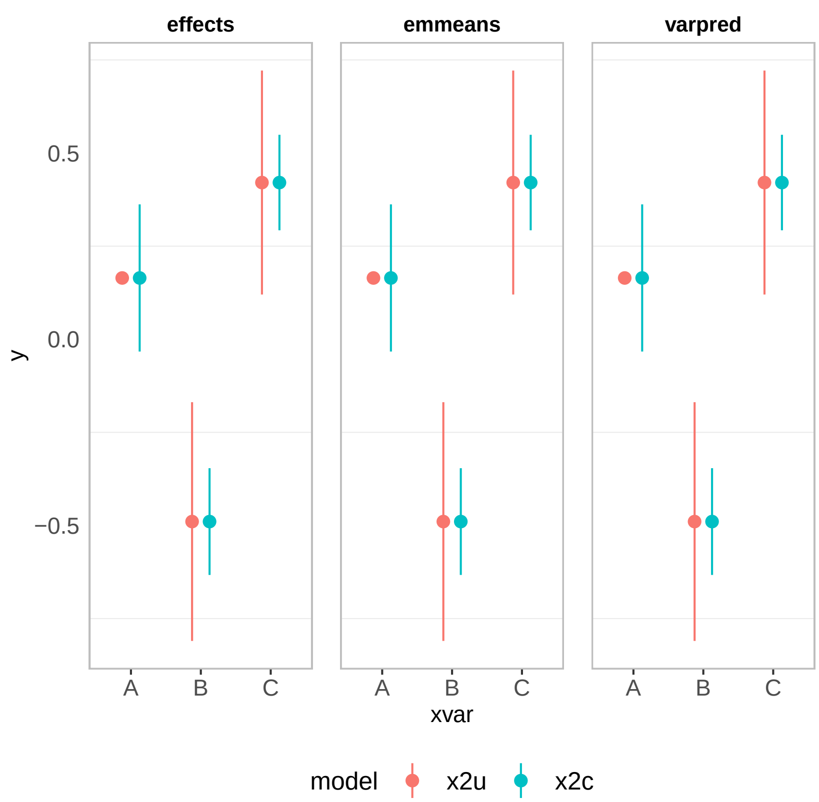


Figure 10: Here, we eliminate uncertainties using zeroed-variance covariance approach. All the three methods give similar estimates for the model with non contrast (`x2u`) to the one with sum to zero contrast (`x2c`), respectively. In comparing, `x2u` with `x2c`, the uncertainties around the baseline category is completely eliminated.

5.1.2 Centered model matrix

Currently, only **varpred** provides a way to center the model matrix.

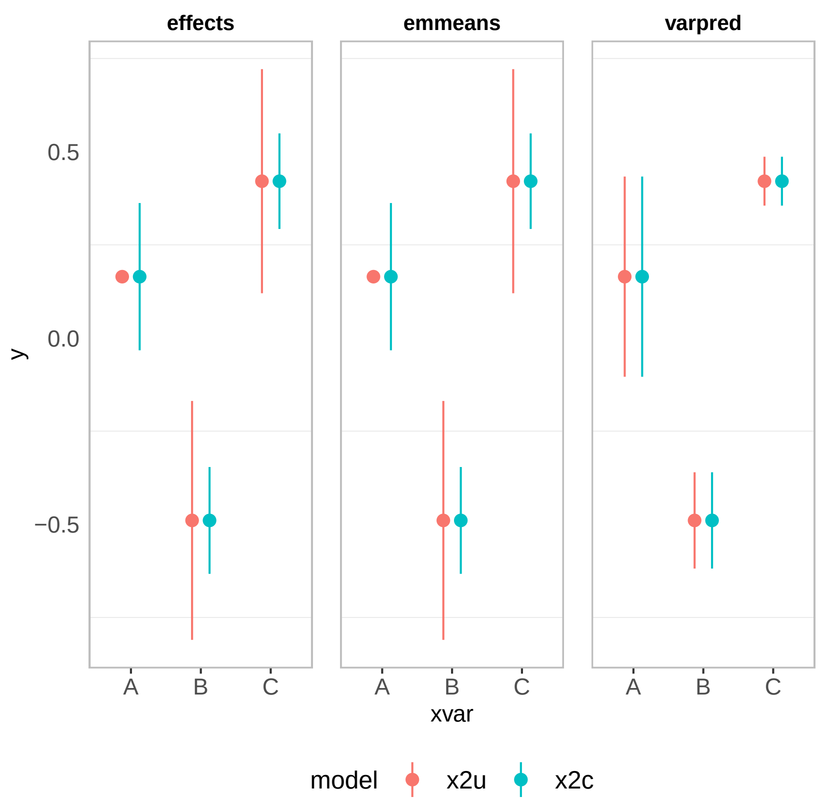


Figure 11: We set `isolate = TRUE` in **varpred** to use centered model matrix in computing the CIs. In this case, predictions for both non contrasted (`x2u`) and contrasted (`x2c`) models are the same in **varpred**. Also, the CIs are slightly different with those in Figure 10. *Steve -> JD: Weighted contrast?*.

References

Stefanie Muff, Leonhard Held, and Lukas F Keller. Marginal or conditional regression models for correlated non-normal data? *Methods in Ecology and Evolution*, 7(12):1514–1524, 2016.