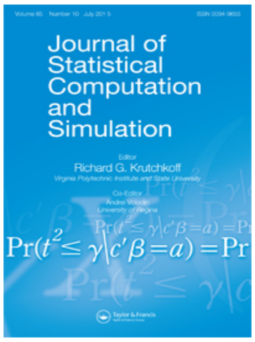
Journal of Statistical Computation and Simulation
Richard G. Krutchkoff
$\Pr(t^2 \leq \gamma | c'\beta = a) = \Pr$

Taylor & Francis
Taylor & Francis Group

Journal of Statistical Computation and Simulation

ISSN: 0094-9655 (Print) 1563-5163 (Online) Journal homepage: https://www.tandfonline.com/loi/gscs20

# A gradient search maximization algorithm for the asymmetric Laplace likelihood

Matteo Bottai, Nicola Orsini & Marco Geraci

**To cite this article:** Matteo Bottai, Nicola Orsini & Marco Geraci (2015) A gradient search maximization algorithm for the asymmetric Laplace likelihood, Journal of Statistical Computation and Simulation, 85:10, 1919-1925, DOI: 10.1080/00949655.2014.908879

**To link to this article:** https://doi.org/10.1080/00949655.2014.908879

*Journal of Statistical Computation and Simulation*, 2015
Vol. 85, No. 10, 1919–1925, http://dx.doi.org/10.1080/00949655.2014.908879

# A gradient search maximization algorithm for the asymmetric Laplace likelihood

Matteo Bottai[a]*, Nicola Orsini[a] and Marco Geraci[b]

[a]*Unit of Biostatistics, Institute of Environmental Medicine, Karolinska Institutet, Nobels väg 13, 17177 Stockholm, Sweden;* [b]*University College London, Institute of Child Health, London, UK*

(*Received 12 February 2014; accepted 23 March 2014*)

The asymmetric Laplace likelihood naturally arises in the estimation of conditional quantiles of a response variable given covariates. The estimation of its parameters entails unconstrained maximization of a concave and non-differentiable function over the real space. In this note, we describe a maximization algorithm based on the gradient of the log-likelihood that generates a finite sequence of parameter values along which the likelihood increases. The algorithm can be applied to the estimation of mixed-effects quantile regression, Laplace regression with censored data, and other models based on Laplace likelihood. In a simulation study and in a number of real-data applications, the proposed algorithm has shown notable computational speed.

**Keywords:** quantile regression; Laplace regression; mixed-effects quantile regression; asymmetric Laplace distribution; direct search optimization algorithms

## 1. Introduction

An increasing number of recently proposed methods for the estimation of conditional quantiles of a continuous outcome variable given covariates make use of the asymmetric Laplace distribution.[1–6] We describe a maximization algorithm for the asymmetric Laplace likelihood and compare its large-sample performance with that of the Frisch–Newton (FN) interior-point method for the estimation of quantile regression. Simple extensions of the presented algorithm have been applied to the estimation of Laplace regression with censored data [7] and mixed-effects quantile regression as an alternative to the Gibbs sampler.[8,9]

Let $y_i$, $i = 1, \ldots, n$, be a sample of continuous variables and $x_i$ be $k$-dimensional vectors of corresponding covariates. We consider the regression model:

$$y_i = x_i'\beta_p + u_i, \tag{1}$$

where the residual $u_i$ follows an unspecified distribution with $P(u_i \leq 0) = p$ for a given probability $p \in (0, 1)$. If model (1) is correct, then $x_i'\beta_p$ is the $p$-quantile of the conditional distribution of $y_i$ given $x_i$. The unknown regression coefficient $\beta_p \in \mathbb{R}^v$ depends on $p$ and can be estimated by minimizing a weighted sum of absolute residuals with computationally efficient methods.[10–12]

*Corresponding author. Email: matteo.bottai@ki.se

It is known [13] that this minimization problem can be seen as the maximization of a likelihood function by assuming that the regression residual $u_i$ follows the asymmetric Laplace density function $f(u_i) = p(1-p)\exp\{u_i(I_{u_i \leq 0} - p)\}$. Here and throughout, $I_A$ denotes the indicator function of the set $A$.

The log-likelihood function of model (1)

$$l(\beta_p) = n\log\{p(1-p)\} + \sum_{i=1}^{n}(y_i - x_i'\beta_p)(I_{y_i \leq x_i'\beta_p} - p) \tag{2}$$

is continuous at all $\beta_p \in \mathbb{R}^v$. Its first derivative, $\partial l(\beta_p)/\partial\beta_p$, is continuous everywhere except at the points $\beta_p$ such that $y_i = x_i'\beta_p$ for at least one observation $i \in \{1, 2, \ldots, n\}$. We define the gradient

$$s(\beta_p) = -\sum_{i=1}^{n} x_i(I_{y_i \leq x_i'\beta_p} - p). \tag{3}$$

The function, $s(\beta_p)$, $\mathbb{R}^v \mapsto \mathbb{R}^v$, equals the first derivative $\partial l(\beta_p)/\partial\beta_p$ at all points of the parameter space where the latter is defined.

## 2. Gradient-search algorithm

In this section, we describe an algorithm for the unconstrained maximization of the log-likelihood function, $l(\beta_p)$, with respect to the parameter, $\beta_p \in \mathbb{R}^v$. Briefly, from a current parameter value, the algorithm searches the positive semi-line in the direction of the gradient $s(\beta_p)$ for a new parameter value at which the likelihood is larger. The algorithm stops when the change in the log-likelihood is less than a specified tolerance. Convergence is guaranteed by the continuity and concavity of the log-likelihood as shown in Section 3.

The algorithm can be summarized by the following steps:

```
(1) Set $k = 0$ and the initial values $\beta_p^0 \in \mathbb{R}^v$ and $\delta^0 > 0$
(2) If $l(\beta_p^k) > l\{\beta_p^k + \delta^k s(\beta_p^k)\}$
    (a) then set $\delta^{k+1} = a\delta^k$
    (b) else if $\{l(\beta_p^{k+1}) - l(\beta_p^k)\} > \epsilon$
         (i) then set $\beta_p^{k+1} = \beta_p^k + \delta^k s(\beta_p^k)$; $\delta^{k+1} = b\delta^k$
        (ii) else return $\beta_p^{k+1}$; stop
(3) Set $k = k + 1$; go to step 2.
```

The algorithm requires setting the following initial values: the starting value of the parameter, $\beta_p^0 \in \mathbb{R}^v$; the initial step-length, $\delta^0 > 0$; the factor for shortening the step-length, $a \in (0, 1)$; the factor for expanding the step-length, $b \geq 1$; and the tolerance for the change in the log-likelihood, $\epsilon > 0$. In addition, the constant $p \in (0, 1)$ and the data $y_i$ and $x_i$ are required to evaluate the functions $l(\beta_p)$ and $s(\beta_p)$.

## 3. Convergence of the algorithm

This section discusses the convergence of the gradient search (GS) algorithm defined in Section 2 and states some properties of the log-likelihood function $l(\beta_p)$ defined in Equation (2).

Definition 1 *Let* $\Omega = \{\beta_p^* \in \mathbb{R}^v : l(\beta_p^*) \geq l(\beta_p), \forall \beta_p \in \mathbb{R}^v\}$ *be the set of maximizers of the log-likelihood function.*

The log-likelihood function $l(\beta_p)$ is a sum of continuous, concave functions and therefore itself continuous and concave. Its continuity and concavity guarantee that a unique maximum exists and that the set of maximizers, $\Omega \subset \mathbb{R}^v$, is compact. The well-known results stated in Lemma 1 is instrumental in establishing convergence of the GS algorithm in Theorem 1.

Lemma 1 *For any sequence of parameters* $\{\beta_p^k\}$ *such that* $l(\beta_p^k) < l(\beta_p^{k+1}) \forall k \in \mathbb{N}$

$$\lim_{k\to\infty} \beta_p^k = \beta_p^* \in \Omega.$$

*Proof* The statement in Lemma 1 follows by the continuity and concavity of the log-likelihood function $l(\beta_p)$ defined in Equation (2). ■

The following Theorem 1 shows that the sequence of parameter values $\beta_p^{k+1} = \beta_p^k + \delta^k s(\beta_p^k)$ iteratively produced by Step 2.b.i of the GS algorithm satisfies the condition in Lemma 1.

Theorem 1 *For any given point* $\beta_p \in \mathbb{R}^v$, *such that* $y_i \neq x_i'\beta_p$ *for all* $i \in \{1, \ldots, n\}$, *there exists a value* $\delta_0 > 0$ *such that* $l(\beta_p) \leq l(\beta_p + \delta s(\beta_p))$ *for every* $0 < \delta < \delta_0$.

*Proof* The directional derivative of $l(\beta_p)$ in the direction $v \neq 0$ evaluated at $\beta_p$ is

$$\begin{aligned} \frac{\partial}{\partial \delta} l(\beta_p + \delta v)\bigg|_{\delta=0} &= \frac{\partial}{\partial \delta} \sum_{i=1}^{n} (y_i - x_i'\beta_p - \delta x_i' v)(I_{y_i \leq x_i'\beta_p + \delta x_i' v} - p)\bigg|_{\delta=0} \\ &= -\sum_{i=1}^{n} x_i' v (I_{y_i \leq x_i'\beta_p} - p)^{I_{y_i \neq x_i'\beta_p}} (I_{x_i' v \geq 0} - p)^{I_{y_i = x_i'\beta_p}}. \end{aligned}$$

Assuming that $y_i \neq x_i'\beta_p$ for all $i \in \{1, \ldots, n\}$, the directional derivative in the direction of the gradient $v = s(\beta_p)$ evaluated at $\beta_p$

$$\left[-\sum_{i=1}^{n} x_i (I_{y_i \leq x_i'\beta_p} - p)\right]' \left[-\sum_{i=1}^{n} x_i (I_{y_i \leq x_i'\beta_p} - p)\right] = \|s(\beta_p)\|^2$$

is non-negative and greater than that in any other direction $v \in \mathbb{R}^v$ such that $\|v\| = \|s(\beta_p)\|$.

If $\|s(\beta_p^k)\|^2 > 0$, then the directional derivative at $\beta_p^k$ in the direction of the gradient $s(\beta_p)$ is positive and there exists a $\delta_0 > 0$ such that the log-likelihood function increases in a $\delta_0$-neighbourhood of $\beta_p^k$ in the direction of $s(\beta_p)$; that is, $\exists\, \delta_0 > 0 : l(\beta_p^k) < l(\beta_p^k + \delta^k s(\beta_p^k)) = l(\beta_p^{k+1}) \forall \delta \in (0, \delta_0)$. Step 2a of the GS algorithm shortens the step-length $\delta^k$ by a factor $a \in (0, 1)$ until $\delta^k < \delta_0$.

If $\|s(\beta_p^k)\|^2 = 0$, then $\|s(\beta_p^k)\| = 0$. The first-order condition $\|s(\beta_p^k)\| = 0$, along with the continuity and concavity of $l(\beta_p)$ over the entire space $\mathbb{R}^v$, implies that $\beta_p^k$ is a maximizer of the log-likelihood, i.e. $\beta_p^k \in \Omega$. ■

Theorem 1 assumes that $y_i \neq x_i'\beta_p^k$ for all observations $i \in \{1, \ldots, n\}$. With the GS algorithm defined in Section 2, the event that $y_i$ is numerically equal to $x_i'\beta_p^k$ for an observation $i$ occurs with probability zero. If it did occur, the GS would move off of it in the direction $s(\beta_p)$ defined in Equation (3).

While meeting the condition of Lemma 1 guarantees that a parameter sequence $\{\beta_p^k\}$ convergences to a maximizer, it does not guarantee that it converges efficiently. Indeed, in an infinite number of iterations, any algorithm that generates a sequence of increasing likelihood values would converge to the maximum. For example, a trivial compass search along the orthogonal directions of $\mathbb{R}^v$ would be a convergent, albeit very inefficient, alternative to the GS. Efficiency of the GS algorithm ensues from its moving from the current parameter value in the direction of maximum local increase.

## 4. A simulation study

We performed a simulation study and compared the proposed GS method with the FN interior point method in large samples. In this case, the latter method has been shown to outperform the Barrodale–Roberts method.[11]

With the statistical program R, we generated 7200 samples, which consisted of 100 samples under each of 72 scenarios obtained from the combination of 3 quantiles, 2 sample sizes, 4 regression models, and 3 distributions for the regression residual. The three quantiles were $p \in \{0.50, 0.75, 0.90\}$. The two sample sizes were $n \in \{10^4, 10^5\}$. The four regression models were (1) $y_i = i$, with $i \in \{1, \dots, n\}$, (2) $y_i = 1 + x_i^{(1)} + x_i^{(2)} + u_i$, (3) $y_i = 1 + x_i^{(3)} + \cdots + x_i^{(10)} + u_i$, and (4) $y_i = -30 + 5000\,x_i^{(11)} + 2x_i^{(12)} + 0.5\,x_i^{(13)} + (0.5 + x_i^{(11)} - 0.5\,x_i^{(12)} + 0.5\,x_i^{(13)})u_i$; where $x_i^{(j)} \sim$ Normal(0,1), $j \in \{1, \dots, 10\}$, $x_i^{(11)} \sim$ Uniform(0,1), $x_i^{(12)} \sim$ Bernoulli(0.5), and $x_i^{(13)} \sim$ T(3). The three distributions of the regression residual were $u_i \sim$ Normal(0,1), $u_i \sim$ T(3), and $u_i \sim$ Lognormal(0,1).

The GS algorithm was implemented as an R function with the following starting values: $\beta_p^0$ equal to the least-squares estimate, $\delta^0$ equal to the sample standard deviation of $y$, $a = 0.5$, $b = 1.25$, and $\epsilon = 10^{-10}$. For the FN method, we used the R function `rq.fit.fnb`, which called compiled Fortran code (quantreg package, version 5.05).

In each generated sample, we estimated the regression coefficients with the GS and the FN algorithm. We compared the methods on the following measures:

(1) 'FN-to-GS time ratio': the ratio between mean CPU time of FN and mean CPU time of GS. In all models, the timing of GS included least-squares estimation for the initial values $\beta_p^0$. For models (2)–(4), it also included the standardization of all covariates and subsequent rescaling of the parameter estimates.

(2) 'Condition on residuals met': the proportion of samples in which the number of negative ($R_-$) and the number of positive ($R_+$) residuals satisfied the condition $R_- \leq np \leq n - R_+$.[14, Theorem 3.4]

(3) 'Max $\log_{10} |l(\beta_p^{\mathrm{GS}}) - l(\beta_p^{\mathrm{FN}})|$': the maximum of the logarithm to base 10 of the absolute difference between the maximum log-likelihood values achieved in each sample.

(4) 'Max $|\beta_p^{\mathrm{GS}}/\beta_p^{\mathrm{FN}} - 1|$': the maximum of the absolute relative difference of the parameter estimates in each sample.

(5) 'Observed bias': the difference between the estimated and the true parameter values averaged over the simulated samples.

The results of the simulation are given in Table 1. The observed bias was virtually zero at all quantiles in all the considered scenarios and it was not reported. The entries in Table 1 are averages over 300 samples comprising 100 replicates for each of the 3 distributions of the regression residual. No notable differences were observed across distributions. The performance of the two methods was comparable with respect to maximum likelihood achieved and proportion of sample in which the condition on residuals was met. The GS algorithm was faster than FN, although its

Table 1. Performance of the GS and FN algorithm for quantile regression estimation.

| | | | Condition on residuals met | | | |
|---|---|---|---|---|---|---|
| Sample size | Quantile | FN-to-GS time ratio | GS | FN | Max $\log_{10} \lvert l(\beta_p^{\text{GS}}) - l(\beta_p^{\text{FN}}) \rvert$ | Max $\lvert \beta_p^{\text{GS}}/\beta_p^{\text{FN}} - 1 \rvert$ |
| Model 1: $y_i = i$, with $i \in \{1, \dots, n\}$ | | | | | | |
| $n = 10^4$ | $p = 0.50$ | 9.59 | 1.00 | 1.00 | −Inf | 0.00 |
| | $p = 0.75$ | 9.96 | 1.00 | 1.00 | −Inf | 0.00 |
| | $p = 0.90$ | 10.09 | 1.00 | 1.00 | −Inf | 0.00 |
| $n = 10^5$ | $p = 0.50$ | 14.01 | 1.00 | 1.00 | −Inf | 0.00 |
| | $p = 0.75$ | 13.70 | 1.00 | 1.00 | −11.44 | 0.00 |
| | $p = 0.90$ | 15.11 | 1.00 | 1.00 | −Inf | 0.00 |
| Model 2: $y_i = 1 + x_i^{(1)} + x_i^{(2)} + u_i$ | | | | | | |
| $n = 10^4$ | $p = 0.50$ | 2.02 | 0.54 | 0.52 | −7.16 | 0.00 |
| | $p = 0.75$ | 2.59 | 0.54 | 0.61 | −6.89 | 0.00 |
| | $p = 0.90$ | 2.41 | 0.52 | 0.54 | −6.20 | 0.01 |
| $n = 10^5$ | $p = 0.50$ | 3.98 | 0.39 | 0.58 | −7.98 | 0.00 |
| | $p = 0.75$ | 4.59 | 0.43 | 0.52 | −8.13 | 0.00 |
| | $p = 0.90$ | 4.13 | 0.46 | 0.57 | −8.21 | 0.00 |
| Model 3: $y_i = 1 + x_i^{(3)} + x_i^{(4)} + x_i^{(5)} + x_i^{(6)} + x_i^{(7)} + x_i^{(8)} + x_i^{(9)} + x_i^{(10)} + u_i$ | | | | | | |
| $n = 10^4$ | $p = 0.50$ | 0.96 | 0.35 | 0.40 | −6.64 | 0.00 |
| | $p = 0.75$ | 1.06 | 0.38 | 0.41 | −6.14 | 0.00 |
| | $p = 0.90$ | 1.04 | 0.37 | 0.35 | −5.85 | 0.01 |
| $n = 10^5$ | $p = 0.50$ | 1.82 | 0.28 | 0.40 | −8.25 | 0.00 |
| | $p = 0.75$ | 2.22 | 0.30 | 0.39 | −7.78 | 0.00 |
| | $p = 0.90$ | 2.28 | 0.29 | 0.43 | −7.55 | 0.00 |
| Model 4: $y_i = -30 + 5000x_i^{(11)} + 2x_i^{(12)} + 0.5x_i^{(13)} + (0.5 + x_i^{(11)} - 0.5x_i^{(12)} + 0.5x_i^{(13)})u_i$ | | | | | | |
| $n = 10^4$ | $p = 0.50$ | 1.01 | 0.42 | 0.53 | −6.49 | 0.00 |
| | $p = 0.75$ | 1.96 | 0.50 | 0.47 | −6.76 | 0.01 |
| | $p = 0.90$ | 2.28 | 0.52 | 0.48 | −6.14 | 0.12 |
| $n = 10^5$ | $p = 0.50$ | 1.16 | 0.17 | 0.45 | −7.30 | 0.00 |
| | $p = 0.75$ | 3.10 | 0.38 | 0.37 | −8.27 | 0.00 |
| | $p = 0.90$ | 4.15 | 0.39 | 0.34 | −6.91 | 0.01 |

Entries are averages over 300 simulated samples, 100 from each of 3 different distributions for the regression residual, $u_i \sim$ Normal(0,1), $u_i \sim$ T(3), and $u_i \sim$ Lognormal(0,1). The covariates are $x_i^{(j)} \sim$ Normal(0,1), $j \in \{1, \dots, 10\}$, $x_i^{(11)} \sim$ Uniform(0,1), $x_i^{(12)} \sim$ Bernoulli(0.5), and $x_i^{(13)} \sim$ T(3).

advantage shrank as the number of covariates increased or the sample size decreased. Optimizing the R programming code and compiling it into machine language might further improve its computational speed.

Following an insightful comment from an anonymous reviewer, we checked sensitivity of the algorithm with respect to its initial parameter values $\beta_p^0$. We re-ran the entire simulation study twice. The first time we set all elements of $\beta_p^0$ equal to 0; the second time we set them equal to 100. While GS took longer to converge with the unreasonable starting values than with the least-squares estimates, it was still generally faster than FN. The final estimates for the parameter $\beta_p$ and achieved maximum likelihood seemed largely unaffected by the choice of the initial values and remained comparable to those reported in Table 1.

## 5. Remarks

The GS algorithm presented in this note is similar to the Newton–Raphson algorithm, in that it moves from the current parameter value in the direction of the gradient, and to direct search methods, in that it adjusts iteratively the step-length multiplier. Unlike methods based on vertex-search in a simplex, as Barrodale and Roberts' method,[10] the algorithm proposed in this paper searches the parameter space without constraints. As Newton–Raphson and other optimization algorithms, GS only approximates the optimal solution within a given tolerance.

Perhaps one of the most useful features of the GS algorithm is its wide applicability. GS may be regarded as a general algorithm for unconstrained optimization of any objective function that is continuous, concave, and first-order differentiable everywhere except at a set of point with measure zero. Unlike Newton–Raphson, GS does not require a second derivative, which is generally unavailable in Laplace-based likelihoods. Thanks to its flexibility, the algorithm can be applied to the estimation of Laplace regression with censored data, mixed-effects quantile regression, and other models based on Laplace likelihood or the optimization of objective function with similar features.

We recommend setting the initial parameter, $\beta_p^0$, at reasonable values. While GS appears to be rather insensitive to the choice of the initial parameter value, reasonable starting values generally improve both convergence speed and accuracy. Least-squares estimates have been satisfactory in our experience. The choice of the convergence criteria and tolerance levels may substantially impact convergence. Stricter criteria may improve maximization accuracy but increase computing time. Generally, we recommend setting $\epsilon$ as small, $a$ as close to 1, and $b$ as large, as computational time and user's patience allow. The values used in the simulation study ($\epsilon = 10^{-10}$, $a = 0.5$, and $b = 1.25$) gave a comparable performance with the FN method. These values have also proved adequate in a number of recent applications of the GS algorithm to the estimation of the coefficients of Laplace regression with censored data.[15–17, among others]

The steps of the GS algorithm listed in Section 2 can be modified in several ways. The following are some examples. The convergence criteria in step 2.b can include checking change from the previous iteration in the parameter estimates or in the signs of the residuals. In step 2.b.i, the step-length can be set to the initial value, $\delta^{k+1} = \delta^0$. The step-length multipliers, $a$ and $b$, may depend on the sample size. When searching for a new parameter value along the direction of the gradient, the algorithm could find the root of the directional derivative by bisection or by considering the sign of the residuals. This would reduce the number of iterations and increase the computational burden at each iteration.

In the simulation reported in Table 1 and in a number of real-data applications, the GS algorithm has shown a remarkable computational speed, which may be a desirable feature when analysing large samples. In small samples (e.g. $n = 100$), the GS, FN, and Barrodale–Roberts algorithms converge so quickly as to make any possible slight differences in computational time inconsequential from most practical standpoints.

## References

[1] Bottai M, Zhang J. Laplace regression with censored data. Biom J. 2010;52(4):487–503.

[2] Farcomeni A. Quantile regression for longitudinal data based on latent Markov subject-specific parameters. Stat Comput. 2012;22:141–152.

[3] Geraci M, Bottai M. Quantile regression for longitudinal data using the asymmetric Laplace distribution. Biostatistics. 2007;8:140–154.

[4] Lee D, Neocleous T. Bayesian quantile regression for count data with application to environmental epidemiology. J R Stat Soc: Ser C Appl Statist. 2010;59:905–920.

[5] Liu Y, Bottai M. Mixed-effects models for conditional quantiles with longitudinal data. Int J Biostat. 2009;5. Article 28.

[6] Yuan Y, Yin G. Bayesian quantile regression for longitudinal studies with nonignorable missing data. Biometrics. 2010;66:105–114.

[7] Bottai M, Orsini N. A command for Laplace regression. Stata J. 2013;13(2):302–314.

[8] Geraci M, Bottai M. Linear quantile mixed models. Stat Comput. 2014;24:461–479.

[9] Geraci M. Linear quantile mixed models: the lqmm package for Laplace quantile regression. J Statist Soft. 2014 (in press).

[10] Koenker R, d'Orey V. Computing regression quantiles. Stat Algorithms. 1987;383–393.

[11] Portnoy S, Koenker R. The Gaussian hare and the Laplacian tortoise: computability of squared-error versus absolute-error estimators. Stat Sci. 1997;12:279–296.

[12] Hunter D, Lange K. Quantile regression via an MM algorithm. J Comput Graph Statist. 2000;9:60–77.

[13] Koenker R, Machado JAF. Goodness of fit and related inference processes for quantile regression. J Amer Statist Assoc. 1999;94:1296–1310.

[14] Koenker R, Basset G. Regression quantiles. Econometrica. 1978;46:33–50.

[15] Orsini N, Wolk A, Bottai M. Evaluating percentiles of survival. Epidemiology. 2012;23:770–771.

[16] Bellavia A, Akerstedt T, Bottai M, Wolk A, Orsini N. Sleep duration and survival percentiles across categories of physical activity. Amer J Epidemiol. 2014;179(4):484–491.

[17] Johannessen A, Skorge TD, Bottai M, Grydeland TB, Nilsen RM, Coxson H, Dirksen A, Omenaas E, Gulsvik A, Bakke P. Mortality by level of emphysema and airway wall thickness. Amer J Respir Crit Care Med. 2013;187(6): 602–608.