

تطبيق نماذج GARCH والشبكات
العصبية الإصطناعية على تحليل تجانس
أخطاء السلسلة الزمنية

بحث مقدم من

صفاء محمد على مصطفى

مدرس مساعد بقسم الإحصاء والرياضيات والتأمين

كلية التجارة – جامعة بورسعيد

ملخص البحث

تختص مشكلة البحث في محاولة حل مشكلة Heteroscedasticity أى عدم ثبات تباين أخطاء السلسلة الزمنية والتي تظهر عادة في السلاسل الزمنية الاقتصادية خصوصاً المالية منها وذلك باستخدام أسلوب الشبكات العصبية الاصطناعية Artificial Neural Networks واستخدام النماذج المعممة للانحدار الذاتى المشروط بعدم تجانس التباين GARCH لحل المشكلة موضوع البحث وذلك بالإعتماد على سلسلة بيانات يومية لمؤشر البورصة المصرية الرئيسى EGX30 فى الفترة من ٢٠١٤/٦/١ إلى ٢٠١٧/٣/٢٩.

وقد تم التخلص من المشكلة محل البحث باستخدام تلك الأساليب وتمت المقارنة بين نتائجهما وخلص البحث إلى أن أسلوب الشبكات العصبية هو الأفضل من حيث سهولة التحليل ودقة التنبؤ و القضاء على مشكلات الأخطاء . كما تم إستخدام الشبكات العصبية فى تحسين دقة التنبؤ لنموذج GARCH وقد تم إثبات هذا التحسن فى القيم المقدرة باستخدام إختبار مربع كاي لجودة التوفيق والذى أثبت عدم وجود أية فروق ذات دلالة إحصائية وأيدت تحسين التنبؤ.

Abstract

The Research mainly trying to solve the problem of Heteroscedasticity which appears in the financial series in particular by Using The method of Artificial Neural Networks and GARCH(Generalized Autoregressive Conditional Heteroskedasticity) Models.

The Research uses a Daily data series of the Egyptian stock exchange main index (EGX30) From 1/6/2014 to 29/3/2017 .

The Research Succeeded in Solving the problem of Heteroscedasticity With the previous methods and compared it's Results .

the Research proves that the Neural Networks was the best method for analyzing and forecasting and solving the problem of Heteroscedasticity andthe Combination of Neural Networks with GARCH models was a successful way to improving the results and enhancing the forecasts.

(١) مشكلة البحث :

من أهم الإفتراضات التي يقوم عليها تحليل السلاسل الزمنية هو فرض ثبات تباين (تجانس) الخطأ Homoscedasticity ولكن إذا كانت الأخطاء لا تتمتع بهذه الصفة فإننا نكون بصدد مشكلة عدم تجانس التباين Heteroscedasticity ويترتب على ذلك فقدان المعلومات المقدرة لبعض خصائصها وفقدان كفاءتها في التنبؤ ، لذا تتمثل مشكلة البحث في محاولة حل مشكلة عدم ثبات تباين أخطاء السلسلة الزمنية باستخدام كلاً من النماذج المعممة للانحدار الذاتي المشروط بعدم تجانس التباين أي نماذج عائلة Generalized Autoregressive Conditional Heteroscedasticity Models (GARCH) وأسلوب الشبكات العصبية الاصطناعية Artificial Neural Networks .

(٢) أهداف البحث :

يهدف البحث بصورة أساسية إلى استخدام أسلوب الشبكات العصبية (ANN) ونماذج (GARCH) في محاولة حل مشكلة عدم ثبات التباين كطريقة حديثة كما توجد أيضاً عدة أهداف فرعية تتمثل فيما يأتي :

- ١- استخدام تلك الأساليب في التنبؤ بعد تحليل السلسلة و محاولة حل المشكلة .
- ٢- مقارنة نتائج الطرق المستخدمة في محاولة حل مشكلة عدم ثبات التباين .
- ٣- استخدام الشبكات العصبية في تحسين دقة التنبؤات الناتجة من نموذج (GARCH)

(٣) أهمية البحث:

الأهمية العلمية : من حيث استخدام البحث لبعض أهم الأساليب الإحصائية المتقدمة وهي أسلوب الشبكات العصبية و نماذج (GARCH) و برامج الحاسب الآلى المتخصصة مثل برنامج EViews وبرنامج MATLAB .

الأهمية العملية : من حيث مجال تطبيق البحث وهو قطاع هام بالسوق المصرى ومؤثر محلياً و دولياً ألا وهي البورصة المصرية حيث تعد بورصة الأوراق المالية المرآة التي تعكس حالة الاقتصاد القومى وقد تعددت الدراسات المتعلقة بالأسواق المالية عالمياً فيما ندرت عربياً.

(٤) فروض البحث :

يقوم البحث على الفروض التالية :

- ١- تستخدم نماذج (GARCH) فى حل مشكلة عدم ثبات التباين بالنسبة لبيانات أسواق الأوراق المالية حيث تأخذ بعين الاعتبار مشكلة عدم تجانس الخطأ.
- ٢- تعتبر الشبكات العصبية (ANN) أفضل من نماذج (GARCH) فى حل مشكلة عدم ثبات التباين بالنسبة لبيانات أسواق الأوراق المالية .
- ٣- الدمج بين الشبكات العصبية ونماذج (GARCH) يودى إلى تحسين دقة التنبؤ .

(٥) حدود البحث :

- يقتصر البحث على بورصة الأوراق المالية المصرية فقط .
- المتغير محل البحث هو سلسلة بيانات المؤشر الرئيسى للبورصة المصرية (EGX30) و الذى يقيس أداء أعلى ثلاثين شركة من حيث السيولة والنشاط فقط.
- تم الإعتماد على البيانات التاريخية للبورصة المصرية وإستخدام سلسلة بيانات يومية للمؤشر (EGX30) فى الفترة من ٢٠١٤/٦/١ إلى ٢٠١٧/٣/٢٩ باستثناء ايام العطل أى باستخدام ٦٩٦ مشاهدة . وقد تم إختيار هذه الفترة باعتبارها بداية فترة استقرار اقتصادى بسبب استقرار الحالة السياسية نسبياً .

(٦) الدراسات السابقة :

• الشركسى (٢٠١١)

" تفسير سلوك مؤشر سوق الأوراق المالية الليبى بإستخدام نماذج GARCH "

هدفت هذه الدراسة إلى بناء نموذج مناسب يفسر سلوك سلسلة الأرباح (العوائد) لسوق الأوراق المالية الليبى كسوق ناشئ وإستخدمت الدراسة سلسلة العوائد اليومية لأسعار الإغلاق لمؤشر السوق الليبى خلال الفترة من شهر أبريل ٢٠٠٨ وحتى شهر يوليو ٢٠٠٩ والتي ظهر فيها أثر ARCH مما يعنى وجود مشكلة عدم تجانس التباين وخلصت الدراسة إلى أن النموذج الملائم هو نموذج GARCH(1,1) .

• الوقدي (٢٠١٤)

"نموذج إحصائي مقترح لتوصيف التقلبات في التباين الشرطي بالتطبيق على المؤشر العام لسوق المال المصري"

هدفت هذه الدراسة للوصول لأفضل نموذج للتنبؤ بمؤشرات البورصة المصرية, EGX30, EGX70, EGX100 بالأخذ في الإعتبار مشكلة عدم ثبات متوسط وتباين السلسلة الزمنية وإستخدمت الدراسة مجموعتين من البيانات هي بيانات يومية وبيانات شهرية للمؤشرات الثلاثة في الفترة من ٢٠٠٦/١/٢ وحتى ٢٠١٢/١٢/٣١ وخلصت الدراسة إلى أنه إذا تواجدت مشكلة الارتباط الذاتي أو مشكلة عدم تجانس الخطأ فإنه من أفضل النماذج المستخدمة في هذه الحالة نماذج GARCH وخصوصاً نموذج EGARCH .

• بن الضب (٢٠١٥)

"إستخدام نماذج GARCH للتنبؤ بالصدمات في البورصات العربية كآلية لإدارة الأزمات"

هدفت هذه الدراسة إلى إبراز أهمية نماذج الإنحدار الذاتي المشروطة بعدم تجانس التباين في نمذجة التقلبات والتنبؤ بها كآلية لإدارة الأزمات والإنذار المبكر واستخدمت الدراسة بيانات يومية لمؤشرات أسهم تسع بورصات عربية وهي: أبوظبي ودبي والبحرين ومصر والكويت والمغرب وعمان وقطر والسعودية في الفترة من ٢٠٠٧ حتى ٢٠١٢ وخلصت الدراسة إلى وجود مشكلة عدم تجانس التباين مما يفرض إستخدام النماذج المشروطة بعدم تجانس التباين وكان النموذج الأفضل لتمثيل معظم المؤشرات هو نموذج EGARCH(1,1) .

• زقاي و عبد القادر (٢٠١٦)

" نمذجة عوائد الأسواق المالية والتنبؤ بمستوياتها المستقبلية بإستخدام نموذج الشبكات العصبية الإصطناعية (دراسة حالة سوق دبي المالي من الفترة ٢٠٠٦/١٢/٤ إلى ٢٠١٥/٢/٣)"

هدفت هذه الدراسة إلى التعرف على أسلوب الشبكات العصبية الإصطناعية ومدى قدرتها على نمذجة عوائد مؤشرات الأسواق المالية والتنبؤ وإستخدمت بيانات السلسلة الزمنية لعوائد مؤشر سوق دبي في الفترة من ٢٠٠٦/١٢/٤ إلى ٢٠١٥/١/٤ ثم التنبؤ للفترة من ٢٠١٥/١/٥ إلى ٢٠١٥/٢/٣ وخلصت الدراسة إلى ما يؤكد كفاءة نموذج الشبكات العصبية الإصطناعية على التنبؤ حيث أعطت الشبكة نتائج موافقة للقيم الفعلية إلى حد كبير .

" قياس التطاير فى السلاسل الزمنية المالية (نماذج ARCH & GARCH)

هدفت هذه الدراسة إلى إختيار أفضل نموذج للتباين المشروط ببواقى نموذج السلاسل الزمنية للبيانات الأسبوعية لأسعار خامات نطف سومو فى ظل حالة عدم التأكد الساندة بأسواق النفط .

وإستخدمت الدراسة عدة نماذج من عائلة نماذج ARCH والتي تتعامل مع تقلبات القيم المالية عبر سلسلة زمنية فى الفترة من ٢٠٠٤ وحتى ٢٠١٦ وخلصت الدراسة إلى أن النموذج الملائم هو نموذج GARCH(1,7) .

(٧) الجانب النظرى للبحث :

(١-٧) نماذج ARCH & GARCH :

لقد كانت نماذج القياس الإقتصادي التقليدية تفترض ثبات تباين الأخطاء، غير أن هذا الفرض يعتبر غير واقعي خاصة عندما يتعلق الأمر بالسلاسل المالية. حيث أن معظم المتغيرات المالية تتميز بالتقلب و عدم ثبات تباين الأخطاء عبر الزمن.

(١-١-٧) نموذج الإنحدار الذاتى المشروط بعدم تجانس الخطأ (ARCH) :

يتخذ النموذج الصورة التالية و يمثل نموذج ARCH(1) : [١٢,٩, ٨,٢]

$$Y_t = X_t\beta + \varepsilon_t$$

$$\varepsilon_t = Z_t\sqrt{h_t}$$

$$h_t = \sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i \varepsilon_{t-1}^2$$

حيث :

$$Y_t \rightarrow N(\varepsilon_t, h_t) \quad \varepsilon_t \rightarrow N(0, h_t) \quad Z_t \rightarrow N(0, 1) \quad , \quad \alpha_0 > 0 \quad , \quad \alpha_1 \geq 0$$

وتسمى المعادلة الأولى معادلة المتوسط وهى عبارة عن دالة فى المتغيرات X والخطأ العشوائى

كما تمثل المعادلة الثانية معادلة التباين المشروط والتي تمثل دالة في أخطاء الفترات السابقة :

$$h_t = \sigma_t^2 = h(\varepsilon_{t-1}, \varepsilon_{t-2}, \dots, \varepsilon_p, \alpha)$$

وتسمى q رتبة النموذج ARCH(q) و α متجه المعلمات المجهولة .

(٢-١-٧) نموذج الإنحدار الذاتي المعمم المشروط بعدم تجانس الخطأ (GARCH)

يتخذ النموذج الصورة التالية ويمثل نموذج GARCH(1,1) : [١٢,٩,٨,٢]

$$Y_t = X_t \beta + \varepsilon_t$$

$$h_t = \sigma_t^2 = \alpha_0 + \sum_{i=1}^q \alpha_i \varepsilon_{t-1}^2 + \sum_{j=1}^p \beta_j h_{t-j}^2$$

حيث : $\alpha_0 > 0, \alpha_1 \geq 0, \beta_t \geq 0$

وفى هذه المعادلة نجد أنها مفسرة بدلالة المتوسط α_0 بدلالة مربعات البواقي المتأخرة لمعادلة

المتوسط (ε_{t-1}) و تعرف بطرف ARCH (ARCH-term) و هي تمثل المعلومات الخاصة بالتذبذب في الفترات السابقة و بدلالة التباين للفترة السابقة σ_{t-1}^2 ويعرف بطرف GARCH و يمكن كتابة معادلة التباين للنموذج (p,q) GARCH على الشكل التالي:

$$h_t = \sigma_t^2 = \alpha_0 + \sum_{i=1}^p \alpha_i \varepsilon_{t-1}^2 + \sum_{j=1}^q \beta_j h_{t-j}^2$$

أى أن q هي رتبة ARCH و p هي رتبة GARCH

(٢-٧) الشبكات العصبية

انظمة تكنولوجية تقوم بتشغيل المعلومات من استقبال للمدخلات واجراء عمليات تشغيل وانتاج المخرجات اعتماداً على مجموعة من الدوال الرياضية التي تقوم بأدق الوظائف بأسلوب يحاكي الشبكات العصبية الطبيعية لدى الانسان فتساعد متخذ القرار على اتخاذ قرارات عالية الدقة .

(٧-٢-١) خطوات التحليل والتنبؤ باستخدام الشبكات العصبية : [٢,١,١٠]

إن التنبؤ باستخدام الشبكات العصبية من الأساليب الحديثة التي لاقت إهتماماً واسعاً في مختلف المجالات واستخدمت بشكل واسع لكونها لا تحتاج إلى شروط صارمة ودقيقة للتنبؤ ويمكن تلخيص عمل الشبكة العصبية وفقاً لطريقة الإنتشار الخلفي في الخطوات التالية :

الخطوة الأولى : إختيار المتغيرات Variable Selection

يجب تحديد المتغيرات وإختيار المشاهدات بحيث تمثل المشكلة محل الدراسة تمثيلاً جيداً .

الخطوة الثانية : معالجة البيانات Data Processing

فيها يتم تحديد طريقة المعالجة من بين ثلاثة إختيارات متاحة للسلاسل الزمنية وهي :

• التنبؤ بقيم السلسلة اعتماداً على القيم السابقة لها وعلى القيم السابقة لسلسلة متغير خارجي آخر (NARX)

Nonlinear Autoregressive with External(exogenous)Input

• التنبؤ بقيم السلسلة اعتماداً على القيم السابقة لها فقط (NAR)

Nonlinear Autoregressive

• التنبؤ بقيم السلسلة اعتماداً على القيم السابقة لمتغير آخر خارجي فقط (NIO)

Nonlinear Input Output

الخطوة الثالثة : تقسيم البيانات إلى مجموعات Divide Data into Sets

١- مجموعة التدريب Training Set مجموعة تعلم وتحديد نموذج البيانات

٢- مجموعة التدقيق Validation Set مجموعة الحكم على مهارة الشبكة

٣- مجموعة الإختبار Testing Set وهي مجموعة لإجراء إختبار نهائي للشبكة

الخطوة الرابعة : تصميم نموذج الشبكة العصبية Neural Network Design

عند تحديد نموذج الشبكة العصبية يجب إختيار مايلي :

- عدد النيورونز للإدخال والذي يساوى عدد المتغيرات المستقلة
- عدد الطبقات الخفية والذي يعتمد على التجربة ودرجة تعقيد المشكلة
- عدد النيورونز فى الطبقة الخفية والذي يتحدد عن طريق التجربة
- نيورون الإخراج والذي عادة يساوى واحد
- تحديد درجة التأخر عند تعديل الأوزان

(٧-٢-٢) طريقة الانتشار الخلفى Back Propagation : [١٠, ٢]

ترجع تسميتها الى طريقة التعلم التى تعتمد على مبدأ تصحيح الخطأ وتبدأ هذه الطريقة بايجاد الفرق بين المخرج المطلوب والفعلى وترجع بهذا الخطأ ارتدادياً من الطبقة الأخيرة الى الطبقات الخفية ثم أخيراً الى طبقة المدخلات وفى أثناء هذا الإرتداد يتم تغيير الأوزان فى الاتجاه الذى يدفع بالخطأ الى النقصان وصولاً الى الصفر وتتم على مرحلتين كما يلى :

المرحلة الأولى : مرحلة الانتشار الأمامى :

وتبدأ بإعطاء أوزان عشوائية للترابط بين خلايا الشبكة ومد الشبكة بإحدى المدخلات المعدة للتدريب بطريقة التدريب الإشرافى وتنتقل البيانات من طبقة المدخلات للمعالجة فى الطبقات الخفية وصولاً لتحديد مخرجات الشبكة ولا يحدث فيها أى تعديل للأوزان وتكون مخرجات طبقة المدخلات هى المدخلات للطبقة الخفية كما يلى:

$$v_j = \sum_{i=1}^n w_{ji}^h x_i$$

$$z_j = f_j^h(v_j) = f_j^h(\sum_{i=1}^n w_{ji}^h x_i)$$

$$\text{net}^o = \sum_{j=1}^m w_{kj}^o z_j$$

$$y = f_k^o(\sum_{j=1}^m w_{kj}^o (f_j^h(\sum_{i=1}^n w_{ji}^h x_i)))$$

المرحلة الثانية :مرحلة الانتشار العكسى (الخلفى) :

هى مرحلة ضبط أوزان الشبكة ويطلق عليها الإنحدار التدريجى (Gradient descent) حيث يتم فيها إعادة إنتشار الإشارة من الخرج إلى الدخل بشكل عكسى ويتم ذلك ابتداء من طبقة المخرجات وتبدأ بمقارنة المخرجات الفعلية (y_k) مع المخرجات المطلوبة (t_k) و ذلك عند التكرار (n) و تحديد قيمة الخطأ كما يلى :

$$e_k(n) = t_k(n) - y_k(n)$$

وتكون دالة الخطأ كالتالى وذلك اعتماداً على مجموع مربعات الخطأ :

$$E(n) = \frac{1}{p} \sum_{k=1}^p e_k^2(n)$$

وبعد حساب دالة الخطأ تبدأ الخطوة الثانية للإنتشار الخلفى وينقل الخطأ للطبقات السابقة ويتم تصحيح الأوزان وتكون الأوزان الجديدة المنقولة من طبقة المخرجات للطبقة الخفية على

$$w_{kj}^0(n+1) = w_{kj}^0(n) + \eta \delta_k(n) z_j(n) \quad \text{الصورة :}$$

وبالمثل تكون الأوزان الجديدة المنقولة من الطبقة الخفية لطبقة المدخلات على الصورة :

$$w_{ji}^h(n+1) = w_{ji}^h(n) + \eta \left(f^h(v_j(n)) \sum_k \delta_k(n) w_{kj}^0(n) \right)$$

ويتم تطبيق هذه الخطوات على كل مدخلات التدريب ولمرات عديدة حتى تصل الشبكة الى أقل خطأ ممكن وتصحح الأوزان بمايدفع بالخطأ للنقصان وصولاً للصفر وعندها تصبح الشبكة جاهزة.

الخطوة الخامسة : معيار التقييم Evaluation Criteria

المعيار المستخدم فى شبكة الإنتشار العكسى لتقييم الخطأ هو مجموع مربعات الخطأ MSE

الخطوة السادسة : تدريب الشبكة Neural Network Training

وفيهما يتم تعليم الشبكة وإيجاد مجموعة الأوزان والتي تحدد أقل قيمة لمربع الخطأ

الخطوة السابعة : الإستكمال Implementation

وهى من أهم الخطوات حيث تختبر الشبكة من حيث قدرة التكيف وإمكانية إعادة التدريب والوصول إلى أقل مربع خطأ عند تغير البيانات .

(٨) الجانب التطبيقي للبحث :

(١-٨) التحليل بإستخدام نماذج عائلة GARCH

(١-١-٨) التحقق من سكون السلسلة :

لكى يتم تسكين السلسلة وإستخدام نماذج عائلة GARCH لابد من من إيجاد سلسلة جديدة وهى سلسلة Returns أو (العوائد) الخاصة بسلسلة EGX وتسميتها EGXR

وفقاً للمعادلة التالية : $EGXR = \text{LOG} (EGX/ EGX(-1))$ ويرسم المنحنى الزمني يتضح سكون السلسلة EGXR ولكي يتم التأكد من ذلك يتم إجراء إختبار ديكي فولر المطور على السلسلة EGXR وإتضح سكون السلسلة حيث جاءت قيمة P أقل من 0.05 كما هو موضح في الجداول (ملحق ١) .

٢-١-٨) إختبار عدم تجانس الأخطاء للسلسلة EGXR

تم إجراء إختبار أثر ARCH على السلسلة EGXR وذلك بعد تقدير معادلة الوسط والنتيجة كما في الجدول (١) وبما أن قيمة p المناظرة لـ NR^2 أقل من 0.05 لذلك نرفض فرض ونقبل الفرض البديل أي أن هناك مشكلة Heteroscedasticity في السلسلة EGXR .

جدول (١) إختبار أثر ARCH في السلسلة EGXR

Heteroskedasticity Test: ARCH			
F-statistic	15.04576	Prob. F(1,671)	0.0001
Obs*R-squared	14.75965	Prob. Chi-Square(1)	0.0001

٣-١-٨) التعرف على النموذج :

تم ترشيح ١٤ نموذجاً من عائلة GARCH على القسم الأول من السلسلة EGXR والمستخدم في التقدير وعدده ٦٧٥ مشاهدة وتمت المقارنة بينها وفقاً لمعيار AIC كما هو موضح في جدول (٢)

جدول (٢) المفاضلة بين نماذج GARCH المرشحة

model	AIC	model	AIC
GARCH (1,0)	-5.5462	TGARCH(1,2)	-5.6218
GARCH (2,0)	-5.5979	TGARCH(2,1)	-5.6112
GARCH(1,1)	-5.6023	TGARCH(2,2)	-5.6167
GARCH(1,2)	-5.6045	EGARCH(1,1)	-5.6008
GARCH(2,1)	-5.6082	EGARCH(1,2)	-5.6074
GARCH(2,2)	-5.6053	EGARCH(2,1)	-5.6081
TGARCH(1,1)	-5.6098	EGARCH(2,2)	-5.6115

(٤-١-٨) تقدير النموذج :

النموذج الأمثل والذي تم إختياره هو النموذج TGARCH(1,2) والذي تم تقديره كما هو موضح في جدول (٣)

جدول (٣) النموذج المقدر TGARCH(1,2)

Variable	Coefficient	Std. Error	z-Statistic	Prob.
C	0.000343	0.000574	0.598191	0.5497
Variance Equation				
C	6.69E-05	1.37E-05	4.888267	0.0000
RESID(-1) ²	0.197690	0.045894	4.307552	0.0000
RESID(-1) ² *(RESID(-1)<0)	-0.079780	0.065550	-1.217083	0.2236
RESID(-2) ² *(RESID(-2)<0)	0.328671	0.084718	3.879582	0.0001
GARCH(-1)	0.418570	0.083426	5.017276	0.0000

(٥-١-٨) تحليل تجانس الأخطاء باستخدام ARCH test:

بالنظر إلى جدول (٤) والذي يوضح نتيجة إختبار أثر ARCH لبواقي النموذج المقدر TGARCH(1,2) نجد أن قيمة P المناظرة ل NR^2 أكبر من 0.05 لذا نقبل فرض عدم القائل بأنه لا يوجد أثر ARCH أي أنه تم التخلص من مشكلة Heteroscedasticity والموجوده في البيانات الأصلية بتطبيق نماذج عائلة GARCH

جدول (٤) نتيجة إختبار أثر ARCH لبواقي النموذج المقدر TGARCH(1,2)

Heteroskedasticity Test: ARCH			
F-statistic	0.000122	Prob. F(1,671)	0.9912
Obs*R-squared	0.000123	Prob. Chi-Square(1)	0.9912

(٦-١-٨) التنبؤ

تم التنبؤ بقيم القسم الثاني من السلسلة EGX والمستخدم في التنبؤ وعدده ٢١ مشاهدة اعتماداً على النموذج الأفضل TGARCH(1,2) كما هي في الجداول (ملحق ١) كما تم الحصول على مقاييس التنبؤ لها كما هي موضحة في جدول (٥)

جدول (٥) مقاييس التنبؤ للنموذج الأمثل (TGARCH(1,2)

MSE	0.929
RMSE	0.864
MAE	0.828
MAPE	6.428

(٢-٨) التحليل باستخدام أسلوب الشبكات العصبية (ANN)

(١-٢-٨) تحديد البيانات (المتغيرات)

الخطوة الأولى فى تحليل الشبكات هى تحديد المتغيرات وهى هنا سلسلة EGX محل الدراسة وهى السلسلة كاملة لتدريب الشبكة وعددها ٦٩٦ مشاهدة وبعد الإنتهاء من التدريب سيتم إيجاد القيم المقدرة للجزء المراد التنبؤ به فى نهاية السلسلة وعدده ٢١ مشاهدة.

(٢-٢-٨) معالجة البيانات

حيث أنه تتوفر لدينا سلسلة زمنية واحدة فقط وهى المعنية بالدراسة ويراد توقع القيم المستقبلية للسلسلة الزمنية $Y(t)$ والذى يعتمد على القيم الماضية من هذه السلسلة فقط فإن هذا الشكل من أشكال الإنحدار الذاتى غير الخطى (Non linear (NAR Autoregressive

(٣-٢-٨) تقسيم البيانات إلى المجاميع

فى هذه الخطوة سيتم تقسيم المدخلات عشوائياً إلى ثلاثة مجموعات :

• المجموعة الأولى : التدريب Training

المستخدمة فى تدريب الشبكة وتحتوى على ٧٠% من بيانات السلسلة وعددها ٤٨٨ مشاهدة

• المجموعة الثانية : الفعالية Validation

المستخدمة للتحقق من تعميم الشبكة وتحتوى على ١٥% من البيانات وعددها ١٠٤ مشاهدة

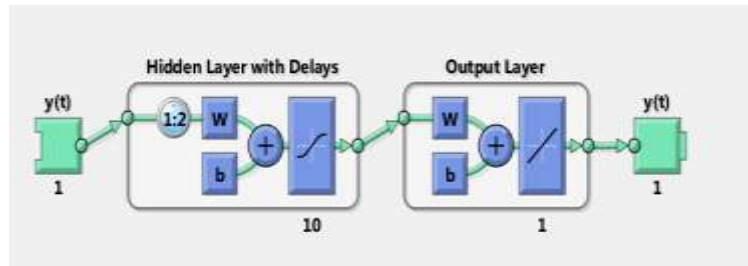
• المجموعة الثالثة : الاختبار Testing

المستخدمة كإختبار مستقل للشبكة وتحتوى على ١٥% من البيانات وعددها ١٠٤ مشاهدة

(٤-٢-٨) بناء الشبكة العصبية

فى هذه الخطوة يتم تحديد نموذج أو بنية الشبكة العصبية

- عدد النيورونز الخاصة بالإدخال والذي يساوى عدد المتغيرات المستقلة والذي يساوى فى هذه الشبكة الواحد (١)
- عدد الطبقات الخفية والذي يعتمد على التجربة و يمكن التحكم فيه حسب نتائج الشبكة عند إعادة التدريب والتي حددت آلياً (إفتراضياً) بطبقة واحدة .
- عدد النيورونز فى الطبقة الخفية والذي يعتمد على التجربة و يمكن التحكم فيه حسب نتائج الشبكة عند إعادة التدريب والذي حدد آلياً (١٠)
- النيورون الخاص بالإخراج والذي يساوى واحد (١)
- تحديد درجة التأخر فى حالة التغذية العكسية لتعديل وتحديث الأوزان فى الشبكة وقد تم تحديده آلياً (٢) والشكل (١) يوضح بناء الشبكة



شكل (١) البنية الأولية للشبكة العصبية NN

(٥-٢-٨) تدريب الشبكة والتنفيذ

- يتم تدريب الشبكة تدريباً إشرافياً وباستخدام طريقة الإنتشار الخلفى للخطأ من خلال الدالة (Trainlm) أو (Livenberg Marquardt Backprobagation) وهى الدالة الأسرع والأكثر إستخداماً فى الشبكات العصبية لتعليم وتدريب الشبكة وتتضمن محددات التدريب مثل معدل التعلم ودالة التنشيط المستخدمة والعدد الأقصى من الدورات وغيرها ويتم حساب مجموعة الأوزان والتي تحدد أقل قيمة لمتوسط مربع الخطأ MSE (ملحق ٢)

- إختبار الشبكة من حيث قدرة التكيف وإمكانية إعادة التدريب للوصول إلى أقل مربع خطأ عند تغير البيانات وفي حالة عدم الحصول على النتائج المرضية يتم إعادة التدريب عدة مرات كما يمكن التغيير في بنية الشبكة حتى يتم الوصول للنتيجة المثلى والتي تضمن أقل قيمة لمتوسط مربع الخطأ.

(٦-٢-٨) النتائج الأولية للشبكة

بعد الإنتهاء من تدريب الشبكة ولأن معيار التقييم المستخدم في شبكة الانتشار الخلفي لتقييم الخطأ هو متوسط مربع الخطأ MSE وكانت قيمته تساوى 0.0151 وهى قيمة جيدة جداً بالمقارنة مع نظيرتها لنموذج GARCH لذا يتم الإتجاه لأختبار تجانس الخطأ بالشبكة

(١-٦-٢-٨) إختبار أخطاء الشبكة الأولية:

أول إختبار سيتم إختباره هو إختبار (archtest) ومخرجاته إما Logical 1 فى حالة وجود مشكلة عدم تجانس البيانات أو Logical 0 فى حالة عدم وجودها و تم تنفيذ هذا الإختبار مرتين الأولى على البيانات قبل التدريب والثانية على الأخطاء بعد التدريب

أولاً : قبل التدريب

```
>> returns=price2ret(EGX);
>> residuals=returns-mean(returns);
>> h=archtest(residuals)

h =

    logical

    1
```

شكل (٢) نتيجة إختبار ARCH قبل تدريب الشبكة NN

ومن الشكل (٢) يتضح وجود مشكلة عدم تجانس الأخطاء حيث كانت النتيجة Logical 1

ثانياً : بعد التدريب

من الشكل (٣) يتضح إستمرار وجود مشكلة عدم ثبات تباين الأخطاء وأنه لم يتم التخلص منها حيث تم تطبيق الأمر على سلسلة الأخطاء الناتجة بعد التدريب (res)


```
>> h=archtest(res)
h =
    logical
     1
```

شكل (٣) نتيجة إختبار ARCH بعد تدريب الشبكة NN

(٢-٦-٢-٨) إعادة تدريب الشبكة لحل مشكلة عدم ثبات تباین الخطأ :

- (الطريقة الأولى) التعديل في درجة التأخر في التغذية العكسية **Number of**
(Feedback Delays)

تم إعادة تدريب الشبكة ولكن بإجراء تعديل في درجة التأخر في حالة التغذية العكسية لتعديل وتحديث الأوزان في الطبقة الخفية بالشبكة وقد تم تغييرها إلى (١) بدلاً من (٢) والتوصل إلى سلسلة أخطاء جديدة وبإختبارها تبين إستمرار المشكلة حيث تم الحصول على Logical 1 مرة أخرى ، ومره أخرى تم إعادة تدريب الشبكة ولكن بإجراء تعديل في درجة التأخر في حالة التغذية العكسية لتعديل وتحديث الأوزان في الشبكة وقد تم تغييرها إلى (٣) بدلاً من (١) والتوصل إلى سلسلة أخطاء جديدة وبإختبارها وتبين عدم وجودها حيث كانت النتيجة Logical 0 كما هو موضح في الشكل (٤)

```
>> h=archtest(res)
h =
    logical
     0
```

شكل (٤) نتيجة إختبار ARCH بعد تدريب الشبكة NN بدرجة تأخر = ٣

- التنبؤ وفقاً للشبكة NN (الطريقة الأولى)

إعتماداً على النتائج النهائية للشبكة العصبية وقيم التنبؤ وفقاً لها و الموضحة في (ملحق ٢) تم الحصول على مقاييس التنبؤ بقيم السلسلة EGX كما هي موضحة في جدول (٦)

جدول (٦) مقاييس التنبؤ باستخدام الشبكات العصبية NN (الطريقة الأولى)

MSE	0.0144
RMSE	0.12
MAE	0.0621
MAPE	0.0032

• (الطريقة الثانية) التعديل في عدد الخلايا العصبية بالطبقة الخفية (Number of Hidden- Neurons)

تم إعادة تدريب الشبكة ولكن بإجراء تغيير في عدد الخلايا العصبية بالطبقة الخفية بالشبكة وقد تم تغييرها إلى (٨) بدلاً من (١٠) والحصول على سلسلة الأخطاء وتم اختبار مشكلة عدم تجانس الأخطاء وإتضح إستمرار وجودها حيث تم الحصول على Logical 1 ثم تم إعادة تدريب الشبكة بتغيير عدد الخلايا العصبية بالطبقة الخفية في الشبكة إلى (١٢) بدلاً من (٨)

وتم اختبار مشكلة عدم ثبات تباين الأخطاء وثبت عدم وجودها حيث كانت النتيجة Logical 0

أى أنه قد تم القضاء على تلك المشكلة باستخدام الشبكة العصبية وهذه هي الطريقة الثانية لحل المشكلة باستخدام الشبكات العصبية .

• التنبؤ وفقاً للشبكة NN (الطريقة الثانية)

تم الحصول على مقاييس التنبؤ بقيم السلسلة EGX اعتماداً على النتائج النهائية للشبكة العصبية NN كما هي موضحة في جدول (٧) والنتائج النهائية موضحة في (ملحق ٢)

جدول (٧) مقاييس التنبؤ باستخدام الشبكات العصبية NN (الطريقة الثانية)

MSE	0.0149
RMSE	0.122
MAE	0.0723
MAPE	0.0017

(٣-٨) الدمج بين الشبكات العصبية وتنبؤات نموذج TGARCH(1,2)

(١-٣-٨) بناء الشبكة (GARCH F - NN)

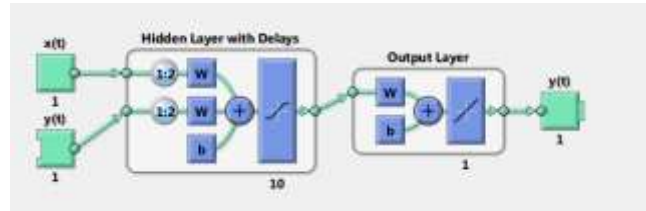
تحديد البيانات سلسلة المدخلات Input هي سلسلة التنبؤات التي تم الحصول عليها من تحليل نموذج **GARCH** وعددها ٢١ مشاهدة والسلسلة الثانية هي سلسلة الهدف **Target** وهي الجزء المراد التنبؤ به في نهاية السلسلة وعدده ٢١ مشاهدة .

معالجة البيانات طريقة التنبؤ الإنحدار الذاتي غير الخطى (NARX) مع متغير خارجي .

تقسيم البيانات إلى المجاميع مجموعة التدريب تحتوى على ٧٠% من البيانات وعددها ١٥ مشاهدة و مجموعة الفعالية تحتوى على ١٥% من البيانات وعددها ٣ مشاهدات و مجموعة الإختبار تحتوى على ١٥% من البيانات وعددها ٣ مشاهدات .

تدريب الشبكة يتم تدريب الشبكة تدريباً إشرافياً وباستخدام طريقة الإنتشار الخلفى للخطأ

بنية الشبكة العصبية بنية الشبكة الموضحة فى الشكل (٥)



شكل (٥) بنية الشبكة العصبية GARCH F - NN

(٢-٣-٨) نتائج الشبكة (GARCH F - NN)

بعد الحصول على النتائج الأولية للشبكة تم إعادة تدريبها للوصول لأقل قيمة ممكنة لمتوسط مربعات الخطأ وتم التوصل للنتيجة النهائية بعد عدة مرات من التدريب $MSE=0.0015$

(٩) نتائج البحث

بناءً على الدراسة التطبيقية توصل البحث للنتائج التالية :

(١) للتخلص من مشكلة عدم ثبات تباين أخطاء السلسلة محل الدراسة تم إتباع عدة طرق كالتالى :

• الطريقة الأولى :

باستخدام نماذج عائلة (GARCH) لتحليل السلسلة محل البحث حيث أن هذه النماذج تأخذ فى إعتبارها عدم ثبات تباين أخطاء السلسلة خصوصاً فى السلاسل المالية لمالها من طبيعة خاصة وقد تم إيجاد سلسلة Returns وتم ترشيح عدة نماذج لتقدير النموذج الأفضل لتمثيل

السلسلة وهو النموذج (1,2) TGARCH وباختبار اخطاء النموذج لم تظهر مشكلة عدم ثبات تباين الأخطاء .

• الطريقتين الثانية و الثالثة:

باستخدام الشبكات العصبية الاصطناعية كطريقة حديثة من طرق الذكاء الاصطناعي قد تم بناء الشبكة ANN لتحليل السلسلة محل الدراسة وأعطت نتائج مرضية بالمقارنة بطرق السلاسل الزمنية وتم إختبار وجود مشكلة عدم تجانس أخطاء الشبكة ولعلاج هذه المشكلة تم إتباع طريقتين وهما :

- التغيير فى درجات التأخر فى التغذية العكسية للشبكة (باستخدام درجة تأخر ٣)
- التغيير فى عدد الخلايا العصبية للطبقة الخفية (باستخدام عدد ١٢ خلية عصبية)

وبالإختبار تبين إختفاء هذه المشكلة بعد إعادة تدريب الشبكة فى الحالتين .

(٢) تم إستخدام الطرق السابقة فى التنبؤ بالقيم المستقبلية للسلسلة EGX والجدول (٨) يوضح المقارنة بين مقاييس التنبؤ لها ويتضح أيضاً أن أفضل الطرق للتنبؤ هى باستخدام الشبكات العصبية حيث حققت أقل قيمة لمتوسط مربع الخطأ يليها نموذج

TGARCH(1,2)

جدول (٨) المقارنة بين طرق التنبؤ بالسلسلة EGX

Model	TGARCH(1,2)	الشبكة NN الطريقة الأولى	الشبكة NN الطريقة الثانية
MSE	0.929	0.0144	0.0149
RMSE	0.864	0.12	0.122
MAE	0.828	0.0621	0.0723
MAPE	6.428	0.0032	0.0017

(٣) إتضح ان أسلوب الشبكات العصبية الاصطناعية هو الأفضل من حيث سهولة التحليل ودقة التنبؤ وأيضاً القضاء على مشكلات البيانات والأخطاء .

(٤) تم استخدام الشبكات العصبية في تحسين دقة التنبؤ حيث تم إدخال قيم تنبؤات نموذج TGARCH(1,2) كخطوة ثانية لتدريبها في الشبكة العصبية وقد تم بالفعل تحسين قيم التنبؤات وقد تم إثبات هذا التحسن في القيم المقدرة باستخدام اختبار مربع كاي لجودة التوفيق والذي أثبت عدم وجود أية فروق ذات دلالة إحصائية وأيدت تحسين التنبؤ وأيضاً أثبت هذا التحسن النقص الحادث في قيم متوسط مربعات الخطأ والجدول (٩) يوضح المقارنة بين هذه التنبؤات قبل وبعد الدمج بالشبكات العصبية

جدول (٩) المقارنة بين تنبؤات النموذج قبل وبعد الدمج بالشبكات العصبية

Model	MSE	χ^2
TGARCH(1,2)	0.929	1.558
(GARCHF – NN)	0.0015	0.060

(١٠) التوصيات :

بناءً على ما توصل إليه البحث من نتائج فإنه يوصى بما يلي :

- استخدام الشبكات العصبية لدراسة سلاسل البيانات المالية والتي تتسم بالتقلبات ولما لها من خصائص تميزها عن باقي أنواع السلاسل الزمنية .
- استخدام الشبكات العصبية لتحسين كفاءة التنبؤ كخطوة تالية لإستخدام نماذج السلاسل الزمنية التقليدية .
- استخدام طرق الدمج لأكثر من أسلوب إحصائي معاً لتحسين دقة التنبؤ .
- يراعى استخدام عينات كبيرة من البيانات عند إستخدام الشبكات العصبية لأنه يحدث فقد بسيط لبعض البيانات أثناء عملية التدريب .

أولا : المراجع العربية

- ١- الحسينى , قصى حبيب و الساعدى , أحمد عبد الأمير (٢٠٠٨)
مقدمة فى الشبكات العصبية الاصطناعية ، العراق ، بغداد، جامعة الإمام جعفر الصادق
- ٢- الشرفاوى , محمد على (١٩٩٦)
الذكاء الاصطناعى والشبكات العصبية ، مصر ، الاسكندرية ، مطابع المكتب المصرى الحديث .
- ٣- الشركسى ، عادل و القبائلى ، أحمد (٢٠١١)
تفسير سلوك مؤشر سوق الأوراق المالية الليبى باستخدام نماذج GARCH ، ليبيا
مجلة جامعة بنغازى العلمية ، العدد الثالث والرابع.
- ٤- العقابى ، عباس لفته كنيهر (٢٠١٧)
قياس التطاير فى السلاسل الزمنية المالية (نماذج ARCH&GARCH)، العراق ،
جامعة واسط ، كلية الإدارة والإقتصاد .
- ٥- الموقع الإلكتروني للبورصة المصرية
<http://www.egx.com.eg/arabic/homepage.aspx>
- ٦- الوقدى ، أحمد فتحى عبد العال (٢٠١٤)
نموذج إحصائى مقترح لتوصيف التقلبات فى التباين الشرطى بالتطبيق على المؤشر العام
لسوق المال المصرى ، كلية التجارة ، جامعة عين شمس.
- ٧- بن الضب ، على (٢٠١٥)
إستخدام نماذج GARCH للتنبؤ بالصدمات فى البورصات العربية كآلية لإدارة الأزمات
، الجزائر، جامعة عين تموشنت.
- ٨- بن أحمد ، أحمد (٢٠٠٧) ،
النمذجة القياسية للإستهلاك الوطنى للطاقة الكهربائية فى الجزائر خلال الفترة (١٩٨٨-
٢٠٠٧)، الجزائر ، جامعة الجزائر.
- ٩- شعراوى ، سمير مصطفى (٢٠٠٥) ،
مقدمة فى التحليل الحديث للسلاسل الزمنية ، مركز النشر العلمى ، السعودية ، جدة

ثانيا : المراجع الاجنبية

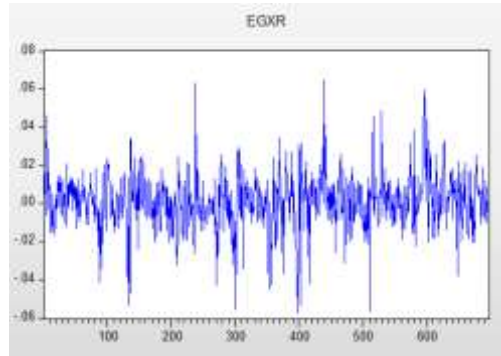
- 10- **Anderson, Dave and Mcneill,George (1992)**
Artificial Neural Networks Technology , Kaman Sciences Corporation , Utica , New York, ,USA.
- 11- **Baum,Christopher(2013)**
Arch and Garch Models ,Boston College ,USA .
- 12- **Bollerslev, Tim (2007)**
Glossary to ARCH (GARCH) , Duke university, USA.
- 13- **B.Yengnanarayana(2005)**
Artificial Neural Networks , Indian Institute of Technology,Madras.
- 14- **Colin,Fife(2000)**
Artificial Neural Networks and Information Theory,University of Paisley, Scotland .
- 15- **Fryzlewicz,Piotr (2007)**
Financial time Series:Arch and Garch Models , University of Bristol ,UK.
- 16- **Hamed , Rania Ahmed (2010) .**
Enhancing the Efficiency of Forecasting Using Time Series Models and Neural Networks , port said university , Egypt.
- 17- **Matei , Marius (2009)**
Assessing Volatility Forecasting : why Garch Models Take the Lead,Romanian Journal of economic Forecasting ,vol(4) .
- 18- **Yim ,Juliana (2002)**

**A Comparison of Neural Net Works With Time Series Models
For Forecasting Returns On A Stock Market Index ,RMIT
University ,Australia .**

الملاحق

المنحنى الزمني للسلسلة EGXR

ملحق ١



إختبار ديكي فولر المطور على السلسلة EGXR

Exogenous	Test Critical Value (5%)	t-statistic	Prob
Intercept	-2.865488	-20.39022	0.0000
Intercept,Trend	-3.416306	-20.43230	0.0000
None	-1.616392	-20.37163	0.0000

جدول (٥- ٢٥) قيم التنبؤ وفقا للنموذج الأمثل (TGARCH(1,2)

Number	Date	Actual Value	Forecasting
1	1/3/2017	11.999	11.942
2	2/3/2017	12.31	11.946
3	5/3/2017	12.5	11.950
4	6/3/2017	12.623	11.954
5	7/3/2017	12.678	11.958
6	8/3/2017	12.735	11.962
7	9/3/2017	12.853	11.966
8	12/3/2017	12.92	11.970
9	13/3/2017	12.929	11.974
10	14/3/2017	12.79	11.979
11	15/3/2017	12.745	11.983
12	16/3/2017	12.984	11.987
13	19/3/2017	13.92	11.991
14	20/3/2017	13.23	11.995
15	21/3/2017	12.905	11.999
16	22/3/2017	12.879	12.003
17	23/3/2017	13.032	12.007
18	26/3/2017	13.108	12.011

19	27/3/2017	12.97	12.016
20	28/3/2017	12.983	12.020
21	29/3/2017	12.988	12.024

ملحق ٢

النتائج الأولية للشبكة العصبية NN

Targets: EGX

Time step: Cell column Matrix column Matrix row

Targets 'EGX' is a 696x1 matrix, representing dynamic data: 696 timesteps of 1 element.

Test Network

MSE 1.51483e-2

R 9.97614e-1

نتيجة الشبكة NN بدرجة تأخر = ٣

Targets: EGX

Time step: Cell column Matrix column Matrix row

Targets 'EGX' is a 696x1 matrix, representing dynamic data: 696 timesteps of 1 element.

Test Network

MSE 1.44796e-2

R 9.97721e-1

نتيجة الشبكة NN بعدد خلايا=١٢

Targets: EGX

Time step: Cell column Matrix column Matrix row

Targets 'EGX' is a 696x1 matrix, representing dynamic data: 696 timesteps of 1 element.

Test Network

MSE 1.49742e-2

R 9.97643e-1

نتائج الشبكة العصبية (GARCH F - NN)

Inputs: GarchF

Targets: target

Time step: Cell column Matrix column Matrix row

Inputs 'GarchF' is a 21x1 matrix, representing dynamic data: 21 timesteps of 1 element.

Targets 'target' is a 21x1 matrix, representing dynamic data: 21 timesteps of 1 element.

Test Network

MSE 1.59677e-3

R 9.71241e-1

قيم التنبؤ وفقاً للطرق الثلاث باستخدام الشبكات العصبية

Number	Date	Actual Value	Forecasting First method-NN	Forecasting Second method-NN	Forecasting GARCH F - NN
1	1/3/2017	11.999	---	---	---
2	2/3/2017	12.31	---	---	---
3	5/3/2017	12.5	---	12.468	12.507
4	6/3/2017	12.623	12.505	12.557	12.567
5	7/3/2017	12.678	12.63	12.653	12.7
6	8/3/2017	12.735	12.677	12.690	12.752
7	9/3/2017	12.853	12.754	12.750	12.8
8	12/3/2017	12.92	12.884	12.882	12.899
9	13/3/2017	12.929	12.896	12.932	12.882
10	14/3/2017	12.79	12.891	12.927	12.856
11	15/3/2017	12.745	12.74	12.763	12.772
12	16/3/2017	12.984	12.788	12.737	12.893
13	19/3/2017	13.92	13.08	13.033	13.083
14	20/3/2017	13.23	13.026	13.091	12.995
15	21/3/2017	12.905	12.913	12.993	12.88
16	22/3/2017	12.879	12.861	12.878	12.909
17	23/3/2017	13.032	12.892	12.873	13.108
18	26/3/2017	13.108	13.07	13.052	13.063
19	27/3/2017	12.97	13.045	13.097	12.984
20	28/3/2017	12.983	12.851	12.933	12.927
21	29/3/2017	12.988	12.989	12.977	13.011

`trainlm` is a network training function that updates weight and bias values according to Levenberg-Marquardt optimization.

`trainlm` is often the fastest backpropagation algorithm in the toolbox, and is highly recommended as a first-choice supervised algorithm, although it does require more memory than other algorithms.

`net.trainFcn = 'trainlm'` sets the network `trainFcn` property.

`[net,tr] = train(net,...)` trains the network with `trainlm`.

Training occurs according to `trainlm` training parameters, shown here with their default values:

<code>net.trainParam.epochs</code>	1000	Maximum number of epochs to train
<code>net.trainParam.goal</code>	0	Performance goal
<code>net.trainParam.max_fail</code>	6	Maximum validation failures
<code>net.trainParam.min_grad</code>	1e-7	Minimum performance gradient
<code>net.trainParam.mu</code>	0.001	Initial mu
<code>net.trainParam.mu_dec</code>	0.1	mu decrease factor
<code>net.trainParam.mu_inc</code>	10	mu increase factor
<code>net.trainParam.mu_max</code>	1e10	Maximum mu
<code>net.trainParam.show</code>	25	Epochs between displays (NaN for no displays)
<code>net.trainParam.showCommandLine</code>	false	Generate command-line output
<code>net.trainParam.showWindow</code>	true	Show training GUI
<code>net.trainParam.time</code>	inf	Maximum time to train in seconds